

Ramona VASILESCU

**SISTEME INFORMATICE DE
CONTABILITATE**

ISBN 978-973-687-710-0

UNIVERSITATEA TIBISCUS TIMIȘOARA
Facultatea de Științe Economice

Lect. drd. Ramona VASILESCU

SISTEME INFORMATICE DE CONTABILITATE

Note de curs pentru uzul studenților de la ÎFR

Editura Eurostampa
TIMIȘOARA
2008

CUPRINS

Cuprins	5
Tema 1. Sistemul informatic – instrument al contabilității.....	7
1.1 Date, informații, sistem informațional.....	7
1.2 Locul și rolul sistemului informatic de contabilitate	11
1.3 Componentele sistemului informatic de contabilitate	13
Tema 2. Caracteristicile programelor de contabilitate	18
2.1 Caracteristici de calitate.....	18
2.2 Constrângeri	23
Tema 3. Realizarea sistemelor informatice de contabilitate	27
3.1 Metodologii de realizare a sistemelor informatice de contabilitate...27	
3.2 Metoda Unified Modeling Language. Prezentare	31
Tema 4. Modelarea sistemului informatic de contabilitate.....	37
4.1 Specificații generale ale metodei Unified Modeling Language	37
4.2 Diagrame utilizate de UML.....	42
Tema 5. Analiza sistemului informatic de contabilitate existent.....	52
5.1 Obiectivele analizei	52
5.2 Elaborarea modelului fizic al sistemului existent.....	54
5.3 Elaborarea modelului logic al sistemului existent.....	63
5.4 Alegerea unui nou sistem informatic de contabilitate	65
Tema 6. Securitatea și controlul sistemelor informatice de contabilitate ...	72
6.1 Securitatea și valoarea informației	72
6.2 Surse de riscuri	75
6.3 Auditul sistemelor informatice de contabilitate.....	79

TEMA 1. SISTEMUL INFORMATIC – INSTRUMENT AL CONTABILITĂȚII

CONȚINUT

- 1.1.** Date, informații, sistem informațional
- 1.2.** Locul și rolul sistemului informatic de contabilitate
- 1.3.** Componentele sistemului informatic de contabilitate

REZUMAT

Orice analiză economică a unei unități economice are la bază informația, privită ca o resursă, și modul în care aceasta este vehiculată. Culegerea, stocarea, prelucrarea, analiza și transmiterea informațiilor sunt activități care trebuie să folosească eficient și eficace resursele informaționale și umane cu scopul obținerii succesului economic. În aceste condiții contabilitatea necesită existența unui sistem informatic de contabilitate performant, care să respecte anumite cerințe organizaționale și legislative.

OBIECTIVE

Tema propusă are ca scop:

- definirea sistemelor informatice de contabilitate;
- stabilirea locului și rolului unui sistem informatic într-o unitate economică.

1.1 DATE, INFORMAȚII, SISTEM INFORMAȚIONAL

Legăturile dintre diversele părți ale unei entități economice trebuie să satisfacă cerințe de calitate și promptitudine care pot proveni fie din interiorul entității economice (cum ar fi cele provenite de la nivelele ierarhice superioare) fie din exteriorul entității economice (de exemplu de la clientul care vrea să știe starea în care se află execuția unei comenzi lansate în producție). Orice analiză economică a unei entități economice are la bază informația și modul în care aceasta este vehiculată, astfel încât degradarea ei să fie minimă și fără pierderi de semnificație. Definițiile informației sunt numeroase și presupun cunoașterea semnificației noțiunilor prin care este definită și, de multe ori, a contextului pentru care a fost definită. Astfel se vorbește despre informație ca despre „comunicare, veste, știre care pune pe cineva la curent cu o situație¹”, „informație genetică”, „informație contabilă” ș.a.m.d. Din punct de vedere economic, informația este privită ca o resursă care poate îmbunătăți raportul cost-eficiență. Bine înțeles că era informaticii în care trăim, aflată la debutul ei, și-a pus deja puternic amprenta asupra modului în care acest raport poate fi modificat, raport care

¹ DEX. Dicționarul explicativ al limbii române, Editura Univers Enciclopedic, București, 1998

este influențat continuu și de nivelul de dezvoltare a tehnologiei hardware și software de la un moment dat. Obținerea informațiilor și prelucrarea lor se face prin intermediul unor sisteme informaționale.

De obicei, oamenii presupun existența unui calculator când aud pentru prima dată sintagma *sistem informațional*. Totuși, un sistem informațional nu este neapărat un sistem computerizat și în fiecare zi vedem exemple de astfel de sisteme informaționale. De exemplu, sunteți beneficiarul unui sistem informațional atunci când doriți să călătoriți cu autobuzul și pentru aceasta cumpărați un bilet. Atunci când prezentați biletul unui controlor, biletul reprezintă suportul informației pe care controlorul o interpretează prin aceea că ați cumpărat dreptul de a călători cu acel mijloc de transport.

Un sistem informațional este parte a unui sistem complet. Un **sistem** este o entitate compusă din părți organizate și care interacționează pentru o funcționare cât mai eficientă. **Subsistemele** sunt părți componente ale sistemului. De exemplu, Facultatea de Științe Economice este un subsistem al sistemului Universitatea Tibiscus.

Un **sistem informațional** se compune dintr-o mulțime de subsisteme intercorelate care lucrează împreună pentru colectarea, prelucrarea, stocarea, transformarea și distribuirea informației pentru planificare, luarea deciziilor și control. Fiecare sistem informațional se poate descompune în trei componente principale: intrările, prelucrările și rezultatele. Intrarea într-un sistem informatic poate fi formată din **date** sau din **informații**. Datele sunt fapte brute, neprelucrate despre evenimente care nu au semnificație în sistem și nu sunt organizate. Datele pot fi totuși organizate într-o manieră în care pot fi utile sau pot primi semnificație pentru sistem. Când datele se organizează astfel încât să aibă semnificație pentru sistem ele devin informație. Rafinarea datelor și informațiilor de-a lungul timpului formează un ansamblu numit **cunoștințe**. Sistemele informaționale prelucrează datele și/sau informațiile (sortare, organizare, calcule specifice) obținând informații care sunt structurate în funcție de cerințele utilizatorilor informației. Aceste cerințe pornesc în general de la scopurile pentru care este folosită informația, de exemplu persoanele de la nivelele de conducere folosesc informația pentru planificare, luare de decizii și controlul activităților organizaționale (decizia cumpărării unui echipament necesită informații despre alternative, costul alternativelor și cerințele referitoare la echipamentul necesar pentru o entitate economică). Informațiile și cunoștințele sunt folosite des în activități de controlul. Contabilii pregătesc bugetele (aceasta este o funcție de planificare) astfel încât managerii să poată compara performanțele actuale cu cele planificate și să poată controla activitățile lor pentru a preîntâmpina schimbările nedorite. Figura 1.1 reflectă modul în care datele, informațiile și cunoștințele (care compun așa numita piramidă informațională) colaborează într-un proces permanent, în care datele pot fi folosite pentru a obține informații și cunoștințe, iar cunoștințele, la rândul lor, pot fi folosite pentru a obține informații și date. Precizăm că figura nu prezintă subordonarea celor trei concepte ci vrea să sugereze un raport cantitativ dintre acestea.

Fluxurile informaționale reprezintă totalitatea informațiilor care se vehiculează între emițătorul de informație și receptor. Sistemul informațional comunică cu mediul său extern prin fluxuri informaționale (de

exemplu rapoartele pentru acționari), iar în interiorul său, subsistemele comunică între ele prin alte fluxuri informaționale.

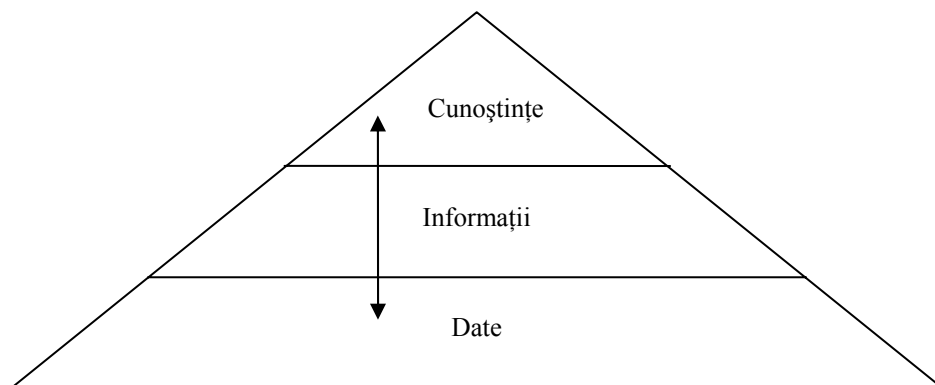


Figura 1.1 Piramida informațională

Sistemele informaționale se studiază în cadrul domeniului în care funcționează, pentru a se evidenția particularitățile specifice, astfel se vorbește de „sistemul informațional de conducere”, „sistemul informațional de marketing”, „sistemul informațional geografic¹” etc. Contabilitatea este în sine un sistem informațional. Este un proces care colectează, stochează, prelucrează și distribuie informații celor care au nevoie de ele. De exemplu, contabilii unei entități economice culeg date despre propria organizație, le prelucrează, obțin rezultate pe care le distribuie sub formă de informații financiare, sau alte tipuri de rapoarte.

Una dintre cele mai cuprinzătoare definiții ale sistemului informațional contabil este cea dată de autorii Gheorghe, Mirela, Roșca, I. Ioan în cartea „*Auditul informației contabile în condițiile utilizării sistemelor informatice*” (pagina 21): „Sistemul informațional contabil este format dintr-un ansamblu de elemente interdependente, orientat spre culegerea, stocarea, prelucrarea, analiza și transmiterea informațiilor privind starea și mișcarea patrimoniului”.

Conceptul de *tehnologie a informației* se referă la totalitatea componentelor software și hardware folosite în sistemele informaționale computerizate. Tehnologia informației a modificat modul în care se lucrează în orice domeniu. În urmă cu câțiva ani, nimeni nu își putea imagina că oamenii vor putea face cumpărăturile de la un magazin virtual „localizat” undeva și accesat prin intermediul Internetului. Comerțul electronic este numai un exemplu din multele moduri în care tehnologia informației influențează viața de zi cu zi dar și cea a afacerilor. Moscove² remarcă faptul că tehnologia informației a avut același impact asupra societății ca și revoluția industrială. În era informației, câțiva muncitori produc și un segment larg de populație angajată este implicată în producția, analiza și distribuția informației, astfel că sistemele informatice au ajuns să joace un rol vital atât în economie cât și în viața fiecăruia. Tehnologia informației afectează orice tip de contabilitate (financiară, de gestiune, managerială). Un sistem informatic de contabilitate este un tip special de sistem, care furnizează informații despre procesele afacerilor și evenimentele care intervin în funcționarea unei unități economice.

¹ www.acad.ro/pro_pri/doc/st_b08.doc

² Moscove, S.A., Simkin, M. G., Bagranoff, Nancy A. – „Core Concepts of Accounting Information Systems”

Dacă la începuturile tehnologiei informației, dependența de sistemele informatice nu se făcea simțită, în ziua de azi nici măcar nu se mai poate imagina ca afacerile să nu folosească sisteme informatice. De la calculatoare se așteaptă să îndeplinească funcțiuni ca: planificarea unei linii de producție, păstrarea evidenței dintr-un depozit, verificarea datelor unui conducător auto etc. Sistemele informatice se folosesc nu numai de către unitățile economice mari care manipulează foarte multe date ci și de către cele mici. Chiar și în țara noastră putem întâlni patroni de microîntreprinderi care țin o evidență contabilă folosind un calculator acasă.

Atunci când este folosită tehnologia informației, de obicei ne referim la acest aspect ca fiind informatic. Definiția din DEX pentru informatic este „știință aplicată care studiază prelucrarea informațiilor cu ajutorul sistemelor automate de calcul”. În cartea „*Bazele computerelor. Hard & soft*”¹, autorii au definit sistemul informatic drept „un sistem informațional care are ca element de culegere, stocare, transmitere și transformare un calculator electronic”.

Dacă vom considera că sistemul informatic este acea parte a sistemului informațional prin care se execută prelucrări automate cu ajutorul sistemelor de calcul, devine evident că sistemul informatic este parte a sistemului informațional. Ținând cont de faptul că prelucrarea datelor în contabilitate se face preponderent automat, prin intermediul programelor specializate, vom considera că sistemele informatice de contabilitate acoperă toate ariile unui sistem informațional de contabilitate.

Sistemele informatice de contabilitate pot furniza o multitudine de tipuri de date și informații: date financiare, date non-financiare, analize rezultate din managementul datelor, informații de căutare sau anticipare, informații despre management, despre acționari, etc.

Sistemele contabile computerizate estompează demarcările dintre sistemele contabilității financiare și ale contabilității manageriale. Multe programe software contabile actuale pot prelua ambele tipuri de date (financiare și non-financiare) și să le organizeze într-o manieră prin care au semnificație atât pentru utilizatorii interni cât și pentru cei externi. Tehnologia informației de azi poate face posibilă obținerea unor rezultate care necesită operații complexe și periodice la intervale scurte de timp (cum ar fi actualizarea la fiecare minut vânzărilor de produse și raportarea acestor vânzări). Aceste rezultate se pot furniza aproape instantaneu prin fax, e-mail, sau pe Internet, pe o pagină specială sau pe propriul site.

Posibilitatea tehnologiei informației de a produce rapid mari cantități de informație poate crea o problemă cunoscută ca *supraîncărcarea informației* ([MOS03]). Prea multă informație și, în mod special, prea multă informație banală, poate copleși utilizatorii informației, iar informația relevantă pentru luarea deciziilor se poate pierde. Contabilii sunt cei care decid natura și sincronizarea informației creată și distribuită printr-un sistem informațional contabil. Influența tehnologiei informației asupra raportărilor financiare primare se face simțită în *furnizarea* informației financiar-contabile. Internetul poate aduce modificări în modul de furnizare a conținutului rapoartelor financiare, sau a disponibilității informației referite în expunerile financiare de bază.

¹ Lupulescu, M. coordonator, Dănăiață, Doina, Muntean, Mihaela – *Bazele computerelor. Hard & soft*, pagina 23

1.2 LOCUL ȘI ROLUL SISTEMULUI INFORMATIC DE CONTABILITATE

Existența sistemelor informatice moderne, de mare viteză, a devenit posibilă după ce calculatoarele s-au răspândit în lumea afacerilor. Istoric, existența sistemelor informatice contabile a început cu informatizarea facturării și a unor operații contabile aferente. În cadrul oricărei unități economice culegerea datelor, prelucrarea lor și obținerea rezultatelor se fac conform unor proceduri organizatorice reglementate fie prin lege (de exemplu componența și structura planului de conturi) fie prin regulamente de ordine internă (de exemplu, stabilirea persoanei și a timpului lansării unei operațiuni de arhivare a datelor).

În figura 1.2 intrările se constituie din date sau informații (care se preiau din documentele justificative), care sunt procesate obținându-se informații pentru planificare, luarea deciziilor și control.

Documentele contabile se clasifică în funcție de rolul lor și de modul de întocmire ([TEA00], pagina 97) în: documente justificative (de evidență primară), registrele contabile (evidență contabilă) și situațiile financiare (documente de sinteză și raportare). Înregistrarea în contabilitate se poate face pentru fiecare document, sau din documentele centralizatoare care cuprind date de aceeași natură dintr-o perioadă oarecare.

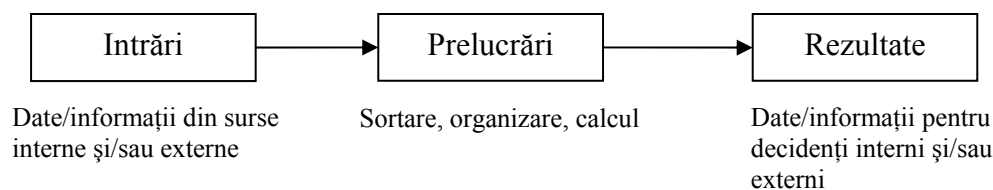


Figura 1.2 Fazele distincte ale funcționării unui sistem
Sursa: [MOS03] pagina 6

Rezultatul prelucrărilor contabile se poate folosi de către nivelele de conducere în cadrul unor procese decizionale, cu observația că „o aceeași informație poate fi percepută cu valori diferite pentru persoane diferite. Aceia care sunt specializați în contabilitate dau importanță mai mare rapoartelor financiare mai mult decât cei care nu au o asemenea pregătire”¹. Informațiile contabile trebuie să îndeplinească următoarele **caracteristici**²:

- *inteligibilitatea* (informațiile pot fi ușor de înțeles și de interpretat);
- *relevanța* (sublinierea aspectelor care pot influența luarea deciziilor);
- *credibilitatea* (informațiile nu conțin erori semnificative, nu sunt tendențioase, nici părtinitoare);
- *comparabilitatea* (informațiile să poată fi comparate prin elemente comune și de aceeași semnificație).

Existența erorilor poate crea incertitudine și luarea unor decizii greșite.

Figura 1.3 reflectă o parte a unui sistem informațional a unei entități economice și scoate în evidență faptul că sistemul informațional de contabilitate este subsistem al sistemului informațional al entității

¹ Hawker, A. „Security and Control in Information Systems: A Guide for Business and accounting”, Routledge 2000, pagina 14

² Teaciuc, M. ș.a. – *Bazele contabilității*, Eurostampa 2000, paginile 7-8

economice. Sistemul informațional de contabilitate acumulează informații de la subsisteme diferite. Pentru ca interacțiunea dintre subsisteme să fie efectivă, este necesar ca fiecare subsistem să „înțeleagă” tipurile de informații generate de subsistemele cu care interacționează.

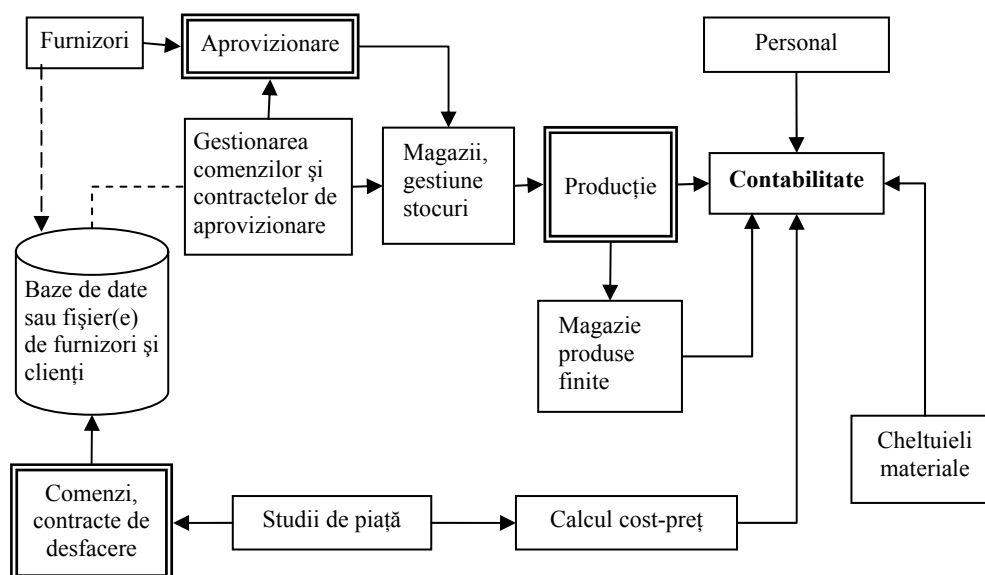


Figura 1.3 Subsisteme informaționale organizate în funcție de activitățile din cadrul unei unități economice

Sursa: Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare” pagina 21

Un sistem informatic de contabilitate tradițional se ocupă în principal de colectarea, prelucrarea și obținerea rezultatelor financiare care se vor transmite către externi (cum ar fi investitorii, creditorii și Ministerul Finanțelor) și către interni (în general structurile de conducere). Un sistem informatic de contabilitate modern se ocupă atât de informațiile non-financiare cât și cu date și informații financiare.

Tradițional, fiecare parte a unei entități economice (Personal, Producție) mențin un subsistem informatic separat și fiecare subsistem își prelucrează propriile date. Această mod de lucru are dezavantajul apariției unor probleme cum ar fi duplicarea datelor pe spații de stocare distincte, culegerea separată a unor aceleași date.

Astăzi entitățile economice consideră că este necesară integrarea funcțiilor lor într-o bază mare de date, sau într-un depozit de date. Această integrare permite managerilor și, cu oarecare extensii, părților externe să obțină informațiile necesare pentru planificare, luarea deciziilor și control, fie că informațiile sunt pentru marketing, contabilitate, sau alte arii financiare ale entității economice¹. Producătorii de produse software au dezvoltat programe care leagă toate subsistemele informatice într-o singură aplicație. Produsele software pentru contabilitate vor fi discutate ulterior.

Rolul sistemului informatic de contabilitate este de a furniza informații importante referitoare la: venituri, urmărirea clienților (facturi emise neîncasate), dinamica încasărilor și plăților, contabilitatea costului

¹ Moscovice, S.A., Simkin, M. G., Bagranoff, Nancy A. – „Core Concepts of Accounting Information Systems”

(calculația costului), cheltuieli, etc. Informațiile furnizate pot îmbrăca forme diverse atât electronic (documente electronice, foi de prezentare electronice, notificări electronice (e-mail), imagini, imagini video, secvențe audio, etc.), fie sub formă „tradițională” (pe suport de hârtie sau folii pentru proiectoare, etc.). Furnizarea informațiilor trebuie să se facă în timp util, cât mai exact, pentru toți destinatarii informației contabile (manageri, personal aparținător altor subsisteme). Un sistem informatic de contabilitate modern este capabil să preia automat datele furnizate de către alte subsisteme.

1.3 COMPONENTELE SISTEMULUI INFORMATIC DE CONTABILITATE

Operațiile contabile care se realizează prin intermediul sistemelor informatice de contabilitate trebuie să ajute la rezolvarea unor probleme specifice ca evidența contabilă a operațiilor pe conturi și calcularea balanțelor contabile. Aceasta se face prin preluarea datelor din documentele de intrare și stocarea lor, prelucrarea datelor și obținerea rezultatelor (vezi figura 1.4). Preluarea datelor se poate face manual (când un operator uman parcurge fiecare document justificativ, operându-l) și/sau automat (când există echipamente periferice de intrare conectate la sistemul informatic). Stocarea datelor presupune existența unui sistem de gestiune a fișierelor și/sau un sistem de gestiune a bazelor de date. Obținerea unui sistem informatic se face urmând câteva etape: analiză, proiectare și implementare, urmărindu-se activitățile din cadrul sistemului informațional aferent și toate fluxurile informaționale care apar, de interes fiind cele care pot fi automatizate prin intermediul calculatoarelor (vezi figura 1.4 care prezintă un exemplu cu activitățile unui sistem informatic).

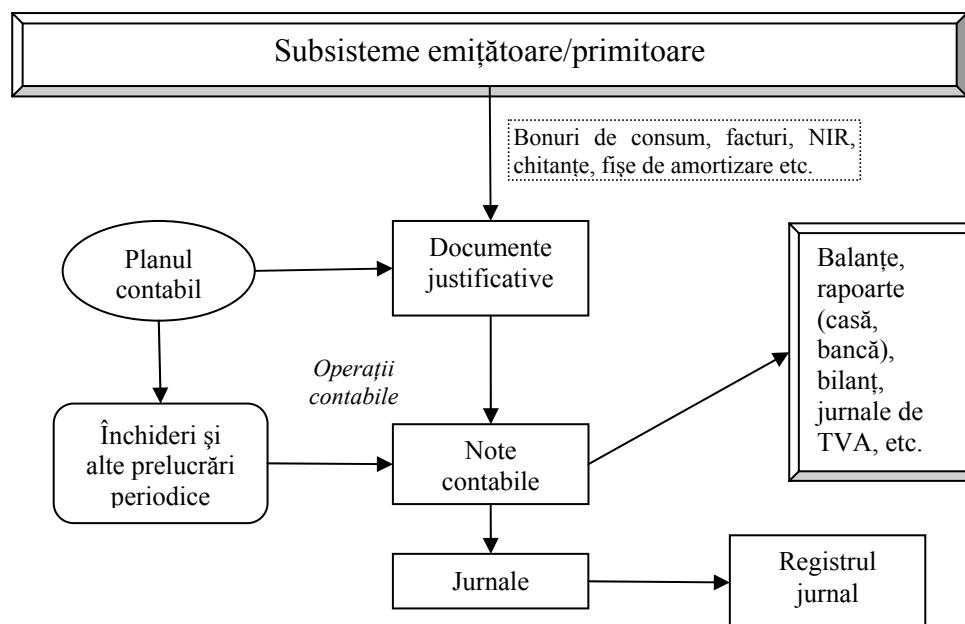


Figura 1.4 Activitățile unui sistem informatic

Se impun câteva observații referitoare la figura 1.4:

1. operațiile contabile cu documentele de intrare justificative înseamnă contarea documentelor: de la subsistemul *Producție*: bonuri de consum, rapoarte de producție; de la *Aprovizionare*: facturi de intrare, NIR (note de intrare-recepție), de la *Vânzări*: facturi emise; de la *Casa, Banca*: chitanțe, foi de vărsământ, ordine de plată; foaie de depunere; etc.;
2. operațiile contabile periodice: închideri lunare și/sau anuale.

Documentele care se pot obține pe baza operațiilor contabile au ca destinatari: nivelului de conducere (decizional), nivelului de control intern și audit.

Componentele unui sistem informatic de contabilitate trebuie să răspundă cerințelor de funcționare descrise până acum și trebuie să fie intercorelate funcțional:

- a. hardware;
- b. software;
- c. comunicație;
- d. baza științifică și metodologică (metodele, procedeele și mijloacele de prelucrare a datelor);
- e. baza informațională, fluxurile informaționale și suporturile de informații;
- f. utilizatorii;
- g. cadrul organizatoric.

Componenta **hardware** se constituie din totalitatea mijloacelor tehnice de culegere, stocare, transmitere și prelucrare automată a datelor. Acestea pot include calculatoare, scannere, case de marcat, dispozitivele de comunicare, dispozitivele de interconectare.

Componenta **software** se constituie din totalitatea programelor și aplicațiilor care realizează funcționarea sistemului informatic. Din această categorie fac parte sistemele de operare utilizate, aplicațiile software de comunicație în rețea, programele de prelucrare în scopul obținerii unor informații contabile, programele de editare de texte și de creare a rapoartelor, etc.

Componenta de **comunicație** se constituie din toate echipamentele și tehnologiile utilizate pentru comunicația datelor între părțile componente ale sistemului informatic.

Baza științifică și metodologică se compune din modelele matematice ale proceselor de contabilitate, din „metodologiile, metodele și tehnicile de realizare a sistemelor informatice”¹.

Baza informațională se constituie din totalitatea fluxurilor informaționale și ale datelor de prelucrat. Unii autori (Lungu I.) includ aici sistemele și nomenclatoarele de coduri. Cum codificarea și utilizarea codurilor este a ajuns un mecanism foarte utilizat chiar și în viața de zi cu zi (de exemplu utilizarea codului CNP) considerăm că acestea sunt de drept o

¹ Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare”, pagina 23

parte importantă a fluxului informațional, având în vedere următoarele: un produs și/sau serviciu se codifică, de cele mai multe ori, printr-un mecanism intern al unității economice; codurile folosite pot fi făcute vizibile (prin liste de selecție, prin afișare lângă preț ș.a), iar cu timpul codurile pot memorate de către utilizatori și folosite cu ușurință într-o manieră directă.

Utilizatorii sunt componenta formată din totalitatea persoanelor angajate în funcționarea sistemului informatic. Se disting două categorii mari de utilizatori: operatorii (de exemplu, cei de la casele de marcat) și informaticienii (cum ar fi analiștii, inginerii de sistem, programatorii).

Cadrul organizatoric este dat de regulamentul de ordine interioară și de actele legislative în vigoare, cum ar fi:

- legea nr. 82/1991 denumită Legea contabilității;
- planul de conturi;
- codul privind Conduita Etică și Profesională a experților contabili și contabililor autorizați din România;
- legea nr. 15/1994 privind amortizarea capitalului imobilizat în active corporale și necorporale;
- ordinul nr. 1826/2003 pentru aprobarea precizărilor privind unele măsuri referitoare la organizarea și conducerea contabilității de gestiune;
- ordinul nr. 1040/2004 pentru aprobarea Normelor metodologice privind organizarea și conducerea evidenței contabile în partidă simplă de către persoanele fizice care au calitatea de contribuabil;
- ordinul nr. 1753/2004 pentru aprobarea Normelor privind organizarea și efectuarea inventarierii elementelor de activ și de pasiv;
- ordinul nr. 1752/2005 pentru aprobarea reglementărilor contabile conforme cu directivele europene.

În **Anexa la Ordinul nr. 58/23 ianuarie 2003** se precizează: „În condițiile utilizării sistemelor informatice financiar-contabile este necesar să fie respectate *Criteriile minimale privind programele informatice utilizate în domeniul financiar-contabil*, prevăzute de Ordinul ministrului finanțelor nr. 425/1998, cu modificările ulterioare” și “Contribuabilii pot edita formularele cu regim special cu ajutorul tehnicii de calcul, în condițiile prevăzute la art. 2 din Ordinul ministrului finanțelor nr. 1.177/1998 privind aplicarea prevederilor art. 1 alin. (4) și (10) paragraful 2 din Hotărârea Guvernului nr. 831/1997”.

În general, sistemele informatice de contabilitate sunt organizate astfel încât să corespundă arhitecturii contabilității din România în care se remarcă trei mari componente ([TEA00]):

1. contabilitatea generală (aprovizionare și furnizori; vânzări și clienți; cheltuieli; venituri; datoriilor; creanțelor; stocuri și obiecte de inventar; mijloace fixe; capital; salarii; operații diverse, de închidere);
2. contabilitatea financiară / de gestiune financiară: trezoreria, investițiile financiare, finanțările – încasări de la clienți, plăți către

furnizori, evidențierea plasamentelor, plata impozitelor și taxelor etc.;

3. controlul prin bugete – elaborarea de bugete și urmărirea acestora prin intermediul contabilității generale.

Din aspectele descrise până acum putem trage concluzia că majoritatea acțiunilor desfășurate în cadrul unei entități economice necesită utilizarea sistemului informatic de contabilitate.

BIBLIOGRAFIE

1. [HAW00] Hawker, A. „*Security and Control in Information Systems: A Guide for Business and accounting*”, Routledge 2000
2. [LUN03] Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare”, Editura Economică, București, 2003
3. [LUP99] Lupulescu, M. coordonator, Dănăiață, Doina, Muntean, Mihaela – *Bazele computerelor. Hard & soft*, Editura Mirton, Timișoara 1999
4. [MOS03] Moscovice, S.A., Simkin, M. G., Bagranoff, Nancy A. – „Core Concepts of Accounting Information Systems”, John Wiley & Sons Ltd, 2003
5. [TEA00] Teaciuc, M. ș.a. – *Bazele contabilității*, Editura Eurostampa, Timișoara 2000
6. ***, <http://www.cdep.ro>, secțiunea „Repertoriul legislativ”

TESTE DE EVALUARE

1. Definiți sistemul informațional:

2. Componentele unui sistem informatic de contabilitate sunt următoarele:

3. Cadrul organizatoric este dat de:

4. Folosind Monitorul Oficial sau alte surse de informare (de exemplu revistele, site-urile și portalurile specializate în furnizarea de informații de contabilitate și colaborare între contabili¹) încercați să găsiți reglementări legislative aplicabile sistemelor informatice de contabilitate, altele decât cele enumerate în cadrul subcapitolului 1.3.

5. Caracteristicile pe care trebuie să le îndeplinească informațiile sunt:

¹ de exemplu: Revista *ContaPlus*, Revista *Contabilitate și informatică de gestiune*, <http://contacafe.ro>, <http://www.contabilul.ro>, <http://www.e-contabilitate.ro>

TEMA 2. CARACTERISTICILE PROGRAMELOR DE CONTABILITATE

CONȚINUT

- 2.1. Caracteristici de calitate
- 2.2. Constrângeri

REZUMAT

Măsura în care un produs software de contabilitate îndeplinește cerințele utilizatorilor depinde direct de totalitatea însușirilor sale pe care acesta le-a dobândit în procesul de producție. Din acest motiv, cunoașterea și determinarea lor este un aspect important al realizării și utilizării programelor de contabilitate.

OBIECTIVE

Tema propusă are ca scop înțelegerea cerințelor de calitate pentru produsele software de contabilitate și ale particularităților acestora care provin din constrângerile impuse.

2.1 CARACTERISTICI DE CALITATE

Informatizarea unităților economice a însemnat crearea unor programe specializate care au trebuit să respecte constrângerile impuse de legislație. Diferențele dintre programe apar la nivelul interfețelor, documentării, asistenței tehnice și al altor servicii. Programele trebuie să respecte anumite principii cum ar fi¹: „prevenirea defectelor; asigurarea faptului că defectele au fost detectate și corectate cât mai curând posibil; stabilitatea și eliminarea cauzelor care produc anumite simptome; audit și conformitate cu standarde și proceduri”.

Prețul programelor de contabilitate diferă în funcție anumite criterii cum ar fi: producătorul, numărul de calculatoare folosite, tehnologia utilizată ș.a. Cele mai ieftine sunt cele care au prețul de achiziție zero lei (Saga C, ContaSQL) iar prețul celor mai scumpe se ridică la câteva mii de lei (Ciel, EasyCont). La prețul de achiziție trebuie adăugat prețul instruirii, asistenței tehnice și al abonamentului pentru actualizările în funcție de modificările legislative. Entitățile economice pot opta pentru achiziționarea unui program de contabilitate prin două metode:

- selectarea unei producător de produse software și crearea unui program de contabilitate special, în funcție de nevoile unității economice;
- selectarea unui program de contabilitate existent.

¹ Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, Editura ASE, București 2003, pagina 4-2

Fiecare dintre aceste metode prezintă avantaje și dezavantaje dar produsul software achiziționat trebuie să răspundă unor cerințe de calitate.

Calitatea produselor software de contabilitate

Măsura în care un produs software de contabilitate îndeplinește cerințele utilizatorilor depinde direct de totalitatea însușirilor sale pe care acesta le-a dobândit în procesul de producție. Cunoașterea și determinarea lor este dificilă din cauza numărului mare și divers al acestor însușiri. De aceea, în practică, se iau în considerare acele însușiri, pe care le vom numi *caracteristici de calitate*, care exprimă direct sau influențează într-un fel sau altul utilizarea produsului.

Calitatea produselor software de contabilitate reprezintă „totalitatea însușirilor tehnice, economice și sociale”¹ și gradul în care ansamblul însușirilor satisfac: nevoia utilizatorilor finali ai produselor, gradul de utilitate și eficiența economică în exploatare. *Gradul de utilitate* al produselor software de contabilitate are în vedere: calitatea proiectării, realizării și execuției; calitatea de conformitate (dintre cerințele utilizatorilor și însușirile actuale ale produselor software); capacitatea de utilizare în rezolvarea problemelor pentru care a fost dezvoltat și capacitatea de mentenanță (măsura în care disfuncționalitățile pot fi reparate).

Caracteristicile de calitate ale produselor software de contabilitate sunt: ergonomia, fiabilitatea, mentenabilitatea, corectitudinea, eficacitatea, stabilitatea, adaptabilitatea, portabilitatea, siguranța în utilizare și claritatea.

Ergonomia este însușirea care exprimă relația directă dintre om și produs prin următoarele caracteristici:

- ușurința exploatării produsului - aceasta se reflectă în produsele software de contabilitate la interfețele care trebuie să fie prietenoase, cu un design plăcut ochiului uman, fără elemente suplimentare care să încarce inutil suprafața de lucru afișată pe ecran;
- securitatea exploatării produsului – programele de contabilitate trebuie să fie prevăzute cu elemente de siguranță cum ar fi: protejarea fișierelor de lucru, imposibilitatea creării unui cont de două ori, imposibilitatea utilizării unui cont nedeclarat, imposibilitatea modificării datelor dintr-o perioadă închisă etc.
- optimizarea solicitărilor fizice și psihice – aplicațiile de contabilitate trebuie să prevadă mecanisme cât mai simple de lucru: alegerea din liste ale denumirilor lungi, completarea automată a informațiilor despre un terț, completarea unor date implicite (cum ar fi data curentă, cota de TVA) etc.;
- consumul de timp pentru obținerea rezultatului – acest consum trebuie să fie cât mai mic pentru oricare dintre operații.

Fiabilitatea reprezintă capacitatea unui produs de a funcționa fără defecțiuni în condiții de lucru bine stabilite. Exprimarea fiabilității folosește

¹ Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, Editura ASE, București 2003, pagina 9-1

noțiunea de *defecțiune* care înseamnă, în fapt, ieșirea din funcțiune și constă în pierderea totală sau parțială, instantanee sau progresivă a capacității de funcționare a produsului. Funcționarea fără defecțiuni constă în menținerea capacității de funcționare a produsului. Astfel dacă se asigură alimentarea continuă cu electricitate, un sistem de operare liber de viruși, o rețea stabilă (dacă este necesară) programele contabile nu au „voie” să își întrerupă execuția sau să lucreze imprecis. Fiabilitatea a dobândit o importanță foarte mare odată cu dezvoltarea tehnologiei și a creșterii complexității produselor de contabilitate.

Mentenabilitatea reprezintă capacitatea ca un produs să poată fi întreținut și reparat într-o anumită perioadă de timp. Ca orice produs și programele de contabilitate pot prezenta defecțiuni care se manifestă în diverse moduri, atât funcțional cât și la nivelul interfeței (o listă atașată unui buton nu se mai deschide, calcularea perioadelor de timp nu respectă anul bisect, ignorarea cifrelor zecimale etc.).

Cum aplicațiile software sunt produse de folosință îndelungată și de importanță mare în cadrul unei unități economice, ele trebuie să fie:

- ușor de menținut în stare de bună funcționare – utilizatorii trebuie să cunoască toate acele condiții suficiente pentru ca această stare să se mențină;
- ușor de întreținut – dacă programul este modular, utilizatorii trebuie să poată adăuga cu ușurință părțile componente care „repară” produsul fără ca aceste componente noi să impiezeze asupra datelor existente; dacă programul nu este modular, producătorul trebuie să prevadă mecanisme simple cu ajutorul cărora utilizatorii să poată înlocui versiunea defectă cu versiunea nouă;
- ușor de reparat – este una dintre cerințele de bază a utilizatorilor, făcând o analogie cu legile lui Murphy, putem spune că orice componentă se defectează tocmai când este cea mai mare nevoie de acea componentă. Producătorii trebuie să ofere modalități simple de contactare din partea utilizatorilor (telefon, e-mail, mesagerie instantanee) și aibă capacitatea de a rezolva orice defecțiune într-un interval de timp cât mai scurt.

Putem trage concluzia că mentenabilitatea unui produs software de contabilitate depinde de următoarele caracteristici:

- accesibilitatea lui – ușurința cu care producătorii pot accesa orice modul constituent al sistemului informatic;
- existența modulelor și/sau versiunilor necesare reparației;
- activitatea de asistență tehnică și întreținere (service) atât în perioada de garanție, cât pe toată durata de utilizare a produsului.

Corectitudinea reprezintă capacitatea unui produs software de contabilitate de a prelucra datele și informațiile și de obține rezultate corecte cantitativ și calitativ, respectând fluxurile transformărilor specificate în documentația ce stă la baza formulării cerințelor utilizatorilor. Un program de contabilitate nu este corect dacă, de exemplu, lucrează intern cu aproximări zecimale de o cifră cunoscut fiind faptul că sunt permise aproximări de cel puțin două cifre.

Eficacitatea reprezintă capacitatea produselor software de contabilitate de a utiliza resursele disponibile cât mai optim oricât de complexă este problema supusă rezolvării. Un program de contabilitate care, astăzi, are prevăzute mecanisme de arhivare numai pe suporturi de memorie de tip dischetă, nu este un program eficient pentru că nu utilizează și alte echipamente periferice disponibile (memoriile de tip „flash”).

Siguranța în utilizare reprezintă capacitatea unui program de contabilitate de a nu permite nici modificarea datelor și nici distrugerea parțială sau totală prin acces neautorizat. Un program de contabilitate care permite ștergerea unui cont sintetic pentru care există înregistrări, nu este un program care oferă siguranță în utilizare. De asemenea, nici dacă permite ca o persoană cu cunoștințe minime de baze de date să poată efectua comenzi de ștergere masivă a datelor din bază. Atragem atenția că distrugerea datelor din cauze pur hardware (distrugerea hard-disk-ului pe care sunt stocate datele) este o caracteristică de calitate a întregului sistem informatic de contabilitate.

Adaptabilitatea reprezintă capacitatea programelor de contabilitate de a permite integrarea unor funcții și/sau componente noi care să mărească performanțele de prelucrare după ce acestea au fost date în funcțiune. Dacă un program de contabilitate extrage o listă a produselor vândute în ultimele două luni, ordonate doar după dată, în cinci secunde, un exemplu de adaptabilitate este adăugarea unei funcții noi de extragere, în doar trei secunde, a acelorași produse ordonate după criterii compuse (cum ar fi: data calendaristică, client și nume).

Portabilitatea reprezintă capacitatea produselor software de contabilitate de a fi funcționale pe mai multe tipuri de calculatoare și/sau sisteme de operare. În practică, atingerea acestei caracteristici se face prin eforturi considerabile de programare și, din acest motiv, referirea la avantajele portabilității poate apare în orice descriere a programelor de contabilitate. De exemplu, dintre avantajele soluției GITS se scoate în evidență că „sistemul este conceput 100% în Java, conferind portabilitate pe orice platformă și sistem de operare. Aplicația rulează pe sistemele de operare Windows 98/ME/2000/XP și Linux având suport pentru baze de date ORACLE, Microsoft SQL Server și MySQL”¹.

Stabilitatea programelor de contabilitate poate fi exprimată:

- din punct de vedere prelucrărilor – reprezintă rezistența programului la modificarea datelor inițiale sau în secvențele ce compun modulele sale; această caracteristică trebuie urmărită atât în timpul procesului de realizare a programelor/modulelor de contabilitate în cadrul testărilor cât și după punerea în funcțiune prin auditul sistemului informatic de contabilitate;
- din punct de vedere software/hardware – reprezintă capacitatea produsului software de contabilitate de a păstra integritatea datelor atunci când apare o întrerupere a disponibilității sistemului. De exemplu, întreruperea bruscă a alimentării cu

¹ <http://www.attosoft.ro>

electricitate nu trebuie să determine apariția unor note contabile incomplete.

Claritatea exprimă măsura în care produsul software de contabilitate este compus numai din instrucțiuni necesare prelucrărilor contabile. Tendința de a crea programe de contabilitate neclare este apanajul producătorilor fără experiența punerii în funcțiune a programelor la mai mulți utilizatori. Claritatea produselor software se poate exprima din două puncte de vedere: al programatorilor și al utilizatorilor finali. Pentru utilizatorul final, un program de contabilitate este neclar dacă interfața este încărcată (explicații inutile, „reclame” referitoare la echipa de programare, legături cu zone care nu sunt de interes, culori șterse sau prea puternice, texte trunchiate, etc.).

Alte caracteristici ale produselor software de contabilitate

Integrabilitatea programelor de contabilitate este o caracteristică urmărită în contextul actual în care se face simțită nevoia utilizatorilor de a utiliza sistemele informatice existente într-un mod unitar. **Integrabilitatea** reprezintă capacitatea produselor software de a fi incluse în sisteme complexe de prelucrare a datelor ([MIH03], pagina 9-6).

În cadrul sistemului informațional contabil putem delimita trei domenii de activitate și anume:

- contabilitatea generală – este acea parte care se ocupă de intrări (de la terți, în gestiune), ieșiri (spre terți, din gestiune), încasări-plăți, operații diverse (salarii, închideri periodice etc.);
- contabilitatea de gestiune – este partea care se ocupă de terți, gestiunea stocurilor, gestiunea lichidităților, inventare, bugete etc.;
- analiza financiară – analiza pe baza bilanțului contabil.

Un sistem informatic de contabilitate poate fi folosit pentru toate cele trei domenii sau numai pentru contabilitatea generală. **Flexibilitatea** este o caracteristică importantă a unui sistem informatic de contabilitate pentru că, în domeniul contabil, schimbările sunt permanente, se pot produce fără avertismente prealabile (conform unei politici guvernamentale cu care contabilii încă nu s-au obișnuit dar în fața căreia s-au resemnat). Cerințele de adaptare rapidă a dus la clasificarea sistemelor informatice de contabilitate în două categorii: aplicații dedicate și aplicații nededicate.

Aplicațiile dedicate sunt acele aplicații care rezolvă punctual o problemă specifică. Dezavantajul acestor aplicații este flexibilitatea scăzută din cauză că este nevoie de intervenția producătorului în cazul în care intervine vreo modificare referitoare la problema aferentă programului. Acest mod de lucru este specific producătorilor de produse software la începuturile existenței lor, programul de contabilitate inițial cunoaște o dezvoltare care îl poate duce la o soluție generală.

Alte aplicații dedicate sunt cele care rezolvă problemele unei anume categorii de probleme; cum ar fi contabilitatea instituțiilor publice. Aceste tipuri de aplicații pot avea o flexibilitate medie provenind din chiar specificul acestui tip de contabilitate.

Aplicațiile nededicate reprezintă un cadru general de rezolvare a unui tip de problemă contabilă, cu flexibilitate mare, orice modificare apărută fie prin reglementări noi legislative fie interne ale unității economice se poate rezolva ușor chiar de către utilizator (de exemplu modificarea sensului unui cont).

Aplicațiile informatice de contabilitate, prin adaptarea în permanență la cerințele pieței, au cunoscut o dinamică pronunțată care s-a datorat atât dezvoltării tehnologice ale componentelor hardware (de exemplu capacitate de stocare, viteză de acces) cât și inovațiilor software (cum ar fi interfețele grafice).

2.2 CONSTRÂNGERI

Programele de contabilitate trebuie create respectând anumite constrângeri care provin din constrângerile legislative (structura planului de conturi, începutul și sfârșitul exercițiului contabil) și constrângerile unităților economice (numărul de calculatoare folosite, numărul de societăți pentru care se face contabilitatea etc.).

Constrângerile administrative provin din resursele necesare pentru utilizarea programelor de contabilitate. Acestea includ: sistemul de operare, necesarul minim de spațiu de memorie externă, necesarul minim pentru capacitatea memoriei interne, diferențele dintre calculatorul server și cele utilizator. Toate aceste aspecte se au în vedere în funcție de dimensiunea unității economice. Astfel o societate comercială cu un număr redus de angajați își poate propune utilizarea unui singur calculator cu un sistem de operare mai puțin performant (cum ar fi Windows 95), cu un hard-disk de capacitate de 400MB. O unitate economică mare s-ar putea să aibă nevoie de o rețea de calculatoare cu un server dedicat stocării datelor, de capacitate mare, la nivelul zecilor de gigabyte, cu programe care să asigure securitatea în rețea și protejarea datelor în cazul unor incidente.

Constrângerile de utilizare a calculatoarelor. Programele de contabilitate se achiziționează monopost (pentru un singur calculator) sau multipost (pentru mai multe calculatoare, atunci când lucrează mai mulți contabili la o unitate economică). Pentru programele de contabilitate multipost se folosesc două arhitecturi¹: partajată și arhitectură client-server.

În cazul arhitecturii partajate, programul de contabilitate și fișierele se găsesc pe un singur calculator (de obicei acesta poartă numele generic „server”), iar contabilii le accesează de la calculatoare conectate în rețea. În cazul arhitecturii client-server, programul de contabilitate este instalat pe fiecare dintre calculatoarele din rețea iar fișierele sunt stocate pe calculatorul numit server.

¹ Boksenbaum, L. – „Informatică de gestiune”, pagina 229

Constrângeri prin numărul de societăți. Contabilitatea se poate ține pentru una sau mai multe unități economice. În acest caz producătorii programelor de contabilitate adoptă două moduri de lucru:

- se permite deschiderea unui număr mare (de ordinul sutelor) fără nicio intervenție a producătorului și/sau fără plată suplimentară; acest mod de lucru este preferat de unitățile economice specializate în furnizarea serviciilor de evidență contabilă;
- se permite deschiderea unei noi societăți contra cost prin intervenția unui angajat al producătorului; acest mod de lucru poate să fie preferat de către organizațiile care administrează una sau mai multe unități economice.

Constrângeri de identificare se folosesc atunci când un utilizator accesează programul de contabilitate sau anumite zone protejate ale acestuia (cum ar fi raportările profiturilor și/sau pierderilor) prin utilizarea unui nume de utilizator și a unei parole. Parolele pot fi generice – caz în care controlul accesului este slab – sau personalizate – caz în care controlul este puternic și permite urmărirea activității unui utilizator în interiorul sistemului informatic.

Constrângerile planului de conturi provin din reglementările legislative care prevăd ca un cont să poată fi creat o singură dată, să nu poată fi șters (dacă a fost folosit cel puțin o dată într-o operație). În mod obișnuit programul de contabilitate este instalat cu un plan contabil predefinit și cu posibilitatea gestionării acestuia. Adăugarea unor conturi noi trebuie să permită stabilirea parametrilor de lucru specifici cum ar fi: codificarea (generică și/sau sintetică; numai numerică, de exemplu 5311, sau și în combinație cu alte caractere, de exemplu 5121.1, 5121.TRA), sensul contului (credit, debit), taxa TVA etc.

Constrângerile de închiderea exercițiului se referă la următoarele aspecte:

- închiderea unui exercițiu trebuie să se facă la începutul anului care îl succede iar soldurile finale ale exercițiului închis devin soldurile inițiale ale exercițiului nou;
- atunci când un exercițiu contabil este închis nu trebuie să se poată efectua modificări;
- să existe posibilitatea deschiderii unui exercițiu închis.

Constrângerile introducerii datelor se referă la modul în care se prelucrează intrările fie prin jurnale fie prin înscrisuri contabile.

Jurnalele se folosesc pentru păstrarea tuturor operațiunilor fie pe categorii de operațiuni (jurnal de vânzări, jurnal de intrări), fie de cont (corelat direct cu un cont cum ar fi cel de casă sau cel de bancă).

Un înscris contabil¹ se identifică prin: data calendaristică, numărul de cont, suma, sensul (debit sau credit) și o explicație. Înscrisurile contabile se folosesc cu scopul de a simplifica introducerea operațiunilor curente fie printr-un „abonament de înscrisuri” (pentru operațiunile care se efectuează

¹ Boksenbaum, L. – „*Informatică de gestiune*”, Editura Economică, București 2002, pagina 230

periodic cu o aceeași sumă) fie prin completarea automată a datelor pentru un cont (cum ar fi contul sintetic al unui terț). Acest mod de lucru necesită o atenție specială atunci când se stabilesc conturile cu prelucrări automate. Dacă nu se cunoaște bine semnificația conturile și funcționarea lor, se pot stabili parametri eronați cur urmări nedorite în balanțele contabile.

Constrângerile operațiilor periodice sunt urmarea faptului că anumite operații trebuie să se desfășoare periodic (obligațiile fiscale, închiderea de lună, închiderea exercițiului etc.). De multe ori, o măsură de siguranță pentru a preveni modificările unei perioade despre care s-a constatat că e validă, se procedează la blocarea perioadei adică nu se mai permite modificarea datelor din perioada respectivă.

Constrângerile personalizate apar ca urmare a formulării cererilor speciale ale unei unități economice și pot fi: modul în care se face imprimarea a unui logo, utilizarea de coduri de bare pentru marcarea documentelor eliberate, comunicarea unor situații în alt mod sau la alte perioade decât cele predefinite, importarea unor date rezultate în urma unor prelucrări externe sistemului informatic de contabilitate și/sau chiar externe unității economice, etc.

Toate constrângerile descrise mai sus au ca punct central informația contabilă. Importanța informației contabile este bine cunoscută și, la fel de bine, este cunoscut faptul că reprezintă peste 40% din informația existentă / utilizată într-o unitate economică. Lipsa informației contabile sau inexactitatea ei poate determina un dezechilibru informațional care să influențeze negativ funcționarea unității economice.

BIBLIOGRAFIE

1. [BOK02] Boksenbaum, L. – „Informatică de gestiune”, Editura Economică, București 2002
2. [MIH03] Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, Editura ASE, București 2003

TESTE DE EVALUARE

1. Enumerați constrângerile pentru programele de contabilitate:

2. Pentru un sistem informatic de contabilitate, planul de conturi:

- a) reprezintă este o constrângere a unui sistem informatic de contabilitate;
- b) nu este însoțit de constrângeri;
- c) este o componentă „închisă” care nu permite modificări.

Care dintre enunțurile a), b) și c) este adevărat. Justificați.

3. Controlul accesului se asigură prin:

- a) constrângerile personalizate;
- b) constrângerile de identificare;
- c) constrângerile administrative.

4. Enumerați domeniile unui sistem informațional de contabilitate:

5. Cerințele de calitate ale unui produs sunt:

6. Mentenabilitatea este:

7. Ergonomia este:

TEMA 3. REALIZAREA SISTEMELOR INFORMATICE DE CONTABILITATE

CONȚINUT

- 3.1. Metodologii de realizare a sistemelor informatice de contabilitate
- 3.3. Metoda Unified Modeling Language. Prezentare

REZUMAT

Realizarea sistemelor informatice se desfășoară în trei etape mari: analiză, proiectare și implementare. Metodologiile de realizare pot fi de ajutor în proiectarea și implementarea sistemelor informatice.

OBIECTIVE

Tema propusă are ca scop înțelegerea modului în care se pot realiza sistemele informatice de contabilitate.

3.1. METODOLOGII DE REALIZARE A SISTEMELOR INFORMATICE DE CONTABILITATE

Sistemele informatice, la fel ca oricare alt produs, au o existență limitată – înlocuirea totală sau parțială fiind necesară din timp în timp. **Ciclul de viață al sistemului informatic** este definit de intervalul de timp $CV = [T1, T2]$ unde $T1$ reprezintă momentul în care s-a decis elaborarea sistemului iar $T2$ reprezintă abandonarea sistemului. Acest interval este abordat metodic în etape ca analiza, modelarea, dezvoltarea, testarea, utilizarea, mentenanța până la retragerea sistemului. Din punctul de vedere al utilizatorului final cele mai importante etape sunt utilizarea și mentenanța.

Ciclu de realizare este dat de intervalul $CR=[T1,T2]$, unde $T1$ reprezintă momentul luării deciziei de realizare iar $T2$ momentul punerii în funcțiune. Acest interval este cel mai important din punctul de vedere al realizatorilor.

Realizarea sistemelor informatice se desfășoară în etape pe baza unor modele și strategii de implementare. Între etapele de realizare a sistemelor informatice există o legătură directă și indestructibilă, calitatea unei etape fiind premisa calității unei etape succesoare. Metodologiile aplicate trebuie să cuprindă toate aspectele referitoare la realizarea unui sistem informatic:

- „etapele/procese de realizare a unui sistem informatic structurate în subetape, activități, sarcini și conținutul lor;
- fluxul realizării acestor etape/procese, subetape și activități;
- modalitatea de derulare a ciclului de viață a sistemului informatic;
- modul de abordare a sistemelor;

- strategiile de lucru/metodele de realizare;
- regulile de formalizare a componentelor sistemului informatic;
- tehnicile, procedurile, instrumentele, normele și standardele utilizate;
- modalitățile de conducere a proiectului (planificare, programare, urmărire) și modul de utilizare a resurselor financiare, umane și materiale etc.”¹

Cum fiecare teorie dezvoltată folosește proprii termeni, în condițiile în care se dorește alegerea celei mai potrivite metodologii de realizare, novicii pot întâmpina greutăți în studiul tuturor metodologiilor.

Clasificarea metodologiilor de realizare a sistemelor informatice de contabilitate

Multitudinea de metodologii se clasifică după criterii diverse cum ar fi: gradul de generalitate, modul de abordare a sistemelor, modelul ciclului de viață ([LUN03]).

Clasificarea metodologiilor de realizare a sistemelor informatice după *gradul de generalitate*:

1. metodologii generale – cu grad înalt de generalitate (SSDAM – Structured System Analysis and Design Methodology, OMT – Object Modeling Technique);
2. metodologii dedicate care se aplică fie numai unor categorii de produse software fie numai unui singur produs software.

Clasificarea metodologiilor de realizare a sistemelor informatice după *modul de abordare* a sistemelor:

1. metodologii cu abordare structurată – sistemul se poate împărți în subsisteme fie din punct de vedere funcțional fie pe baza grupării logice a datelor (STRADIS – Structured Analysis and Design Information Systems, YSM – Yourdon Systems Methods);
2. metodologii cu abordare orientată obiect – folosesc conceptele tehnologiei orientată obiect (UML – Unified Modeling Language, OOD – Object Oriented Design, OOA – Object Oriented Analysis, OOSD – Object Oriented Structured Design)

Clasificarea metodologiilor de realizare a sistemelor informatice după *modelul ciclului de viață*:

1. metodologii cu model în cascadă – etapele se parcurg succesiv, la terminarea unei etape se poate reveni la o etapă anterioară (vezi figura 3.1);

¹ Lungu, I., Sabău, G., Velicanu, M. – „Sisteme informatice. Analiză, proiectare și implementare”, Editura Economică, București 2003, pagina 81

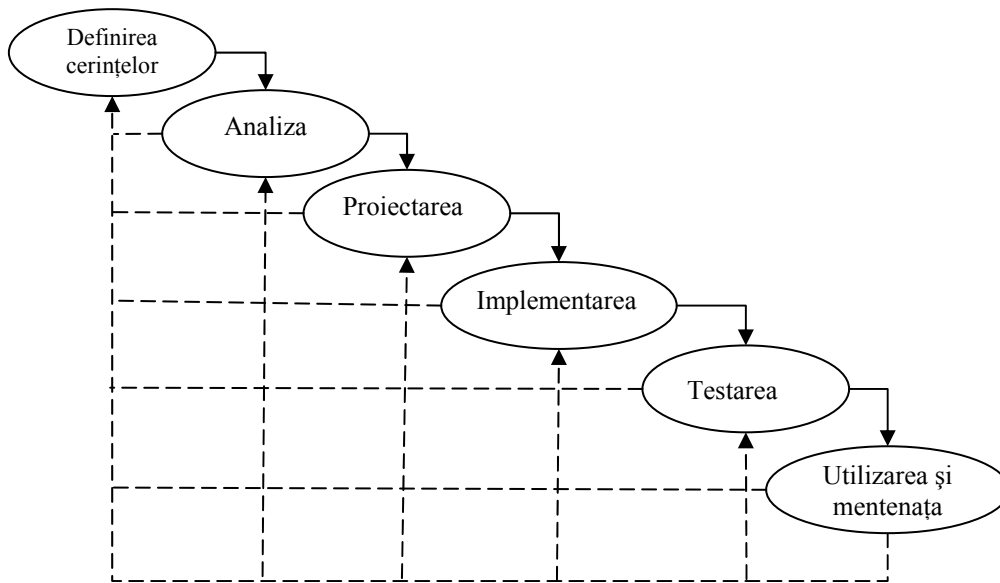


Figura 3.1. Etapele modelului în cascadă

Sursa: Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare” pagina 87

2. metodologii care folosesc modelul prototipului – se elaborează o primă versiune simplificată cu funcționare minimă care este urmată de versiuni succesive îmbunătățite până se atinge funcționarea completă conform cerințelor beneficiarului (vezi figura 3.2);

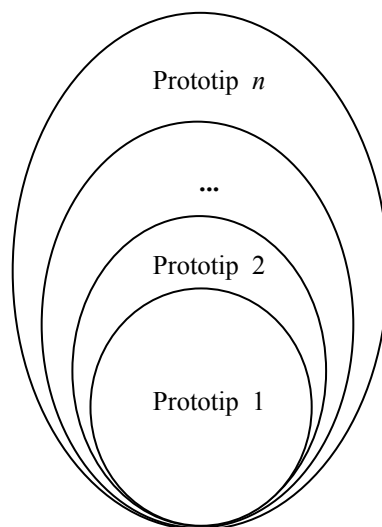


Figura 3.2. Modelul prototipului

Sursa: Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare” pagina 87

3. metodologii incrementale – se folosesc când sistemele informatice se pot pune în funcțiune modular prin realizarea subsistemelor. Etapele acestor metodologii sunt cele ale modelului în cascadă cu deosebirea că proiectarea, implementarea, testarea și utilizarea și întreținerea se realizează separat, pentru fiecare subsistem în parte;

4. metodologii evolutive – se folosesc în cazul realizării sistemelor complexe; se face o descompunere în subsisteme de complexitate redusă și pentru fiecare subsistem se realizează un sistem informatic. În finalul procesului toate subsistemele informatice se assemblează.

Clasificarea metodologiilor de proiectare a sistemelor informatice pe baza *metodelor folosite* ([ORZ05, pagina 170]):

- metodologii bazate pe metode de proiectare „pe măsură” sau la comandă – sistemele informațional se analizează pas cu pas, cu reveniri la pașii anteriori, abordarea problemelor făcându-se de la general spre detaliat;
- metodologii bazate pe metode de proiectare „în serie” – întâi se realizează sistemul informatic pentru o unitate economică pilot; acest sistem se extinde în funcție de prin adaptare și integrare în funcție de specificul și/sau domeniului unității economice;
- metodologii bazate pe metode de proiectare automată – sistemul se realizează prin instrumente software de asistare cu ajutorul calculatorului;

Un aspect comun pentru toate metodologiile descrise este acela că trecerea de la o etapă la alta se face numai după o analiză a activităților, și, pentru fiecare activitate, se stabilesc regulile, standardele de calitate și instrumentele și tehnicilor utilizate.

Modelul în cascadă

Fiecare etapă a modelului în cascadă (vezi figura 3.1) are un scop bine stabilit și se bazează pe rezultatele etapei precedente ([MIH03]):

- *definirea cerințelor* – această etapă are ca scop faptul că trebuie să identifice cerințele de realizare;
- *analiza* – această etapă are ca scop descrierea cerințele de funcționare într-un document;
- *proiectarea* – are ca scop stabilirea arhitecturii sistemului informatic și se face în două sub-etape: proiectarea de ansamblu (se rafinează cerințele de funcționare și restricțiile de funcționare, se stabilește arhitectura produsului software etc.) și proiectarea de detaliu (se specifică algoritmi, modulele, interfețele, fluxurile de date, fluxurile de control etc.);
- *implementarea* – această etapă are ca scop realizarea produsului software care poate fi compus din module, componente speciale, programe utilitare etc.;
- *testarea* - fiecare element constituent cât și întregul sistem informatic de contabilitate trebuie testat; testarea este o activitate importantă care trebuie să se desfășoare înainte de punerea în funcțiune la utilizatorul final;
- *utilizarea și mentenanța* – este etapa în care sistemul informatic este instalat la utilizatorul final și pus în funcțiune; de obicei, aceasta se face modular împreună cu testarea fiecărei componente la locul de muncă. Testarea se face atât la nivel funcțional cât și la nivelul interfețelor – utilizatorul final poate avea obiecții referitoare la designul formularele de introducere a datelor, la listele afișate, la

rapoartele generate.

În cadrul fiecărei etape se elaborează documentația etapei care se constituie ca un rezultat al etapei. Dintre aceste documente amintim: „Proiectul de ansamblu”, „Proiectul de detaliu”, „Specificația de testare” (precizează planul de testate, mediul de test, cazurile și procedurile de testare), „Raportul de testare”, „Manualul de utilizare” etc.

3.2 METODA UNIFIED MODELING LANGUAGE. PREZENTARE

Modelarea este activitatea prin care se descrie un sistem prin intermediul unui ansamblu e notații specifice. UML (acronim pentru Unified Modeling Language) este un limbaj de modelare care dispune de un sistem de notații, de principii și procedee care pot folosi în procesul de abstractizare și, foarte important în dezvoltarea sistemelor informatice, dispune de mecanisme prin care să poată fi extins pentru a putea fi folosit în cazul unor sisteme informaționale speciale. UML este un limbaj de modelare care a fost standardizat de către grupul OMG¹ pe baza unor metode de modelare existente în momentul standardizării (OML – Open Modeling Language, metoda Booch, OMT – Open Method Technique, OOSE – Object Oriented Software Engineering).

Concepte utilizate

Câteva dintre aceste metode se folosesc în reprezentări de: obiecte (*object*), clasa de obiecte (pe scurt *clasă*), abstractizare, încapsulare, moștenire și polimorfism.

Obiectul poate fi considerat un model abstract al oricărei entități fizice sau ne-fizice. Un obiect se caracterizează prin: identitate, stare și comportament. Identitatea este unică și se poate cuantifica printr-un identificator unic (numeric și/sau text) prin care obiectele se diferențiază între ele. *Factura număr 2254/22.09.2007* și *Factura număr 2264/22.09.2007* sunt două obiecte distincte care se identifică în mod unic în anul 2007 prin număr 2254 respectiv 2264. Pe o perioadă de mai mulți ani, facturile se identifică în mod unic prin ansamblul format din numărul facturii și din data calendaristică 2254/22.09.2007 respectiv 2264/22.09.2007.

Unui obiect i se atașează un set de proprietăți (atribute) care conțin informații despre acesta. De exemplu *Beneficiar* este o proprietate a unei facturi care poate lua valori ca „SC Urania SRL”, „SA Pădurea de Aramă”, „SC AmiciiContab SRL” etc. *Valoarea totală* este o altă proprietate a unei facturi care poate lua valori numerice.

Starea unui obiect este reprezentată de toate valorile interne ale proprietăților. În momentul în care se modifică starea unui obiect, identificatorul acestuia nu se modifică. De exemplu, fie factura 2254 care

¹ Object Management Group

are *Valoarea totală* de 2345,56 lei pentru 3 produse, dacă *Valoarea totală* se modifică la 3443,76 lei pentru N+1 produse, numărul de factură rămâne nemodificat.

Comportamentul este dat de mulțimea operațiilor care se efectuează de către obiect atunci când se acționează asupra lui; „clientii” obiectului (alte obiecte care îl utilizează) emit cerințe către obiect iar obiectul „răspunde” prin setul de operații care îi este atașat; de exemplu, obiectul factura 2254 are atașată operația *CalculeazăTVA* care este folosită în operația de contare a TVA. Mulțimea operațiilor se compune din *metode* care din punct de vedere programatic pot fi funcții și/sau proceduri.

Clasa de obiecte, pe scurt *clasa*, grupează obiectele cu aceeași structură (adică aceleași proprietăți) și același comportament. Clasa *Facturi* grupează toate obiectele *factură* care se identifică în mod unic printr-un număr, au aceleași proprietăți (beneficiar, valoare totală etc.). Desigur, la o analiză mai atentă, descoperim că putem folosi două clase *FacturiIntrare* și *FacturiEmise*. Într-un sistem informațional, clasa se identifică în mod unic printr-un nume pentru care se recomandă să se folosească un grupaj de unu sau mai multe cuvinte/prescurtări care să aibă legătură directă cu obiectele din lumea reală pe care le modelează. Pentru o analiză serioasă, ar fi de-a dreptul ciudat ca pentru facturile emise să se modeleze clasa „PiticiPlăți” sau „HârtiiBicolore”. Cum o clasă este o abstracțiune, care descrie toate caracteristicile comune ale unui grup de obiecte, **clasa nu reprezintă un obiect**. Un obiect se obține dintr-o clasă prin **instanțiere**. Putem spune că o **instanță** reprezintă un obiect al clasei care se distinge de alte instanțe ale clasei prin valorile diferite ale atributelor/proprietăților.

Abstractizarea este un proces pe care omul îl folosește, conștient sau nu, în permanență pentru a extrage ceea ce este esențial din lumea înconjurătoare. Fiind un proces subiectiv, care se face diferit de către oameni diferiți (prin vârstă, cultură, nivel de educație), abstractizarea devine cel mai sensibil proces care se utilizează în informatică pentru modelarea sistemelor tocmai pentru că experiența și puterea personală de abstractizare a fiecăruia dintre cei implicați în modelare sunt detalii care pot provoca succesul sau insuccesul unui sistem informatic de contabilitate. Cu toate că în viața de zi cu zi fiecare se descurcă în abstractizarea lumii reale, a explica unui neavizat ce este esențial în contabilitate (ce este un cont, cum funcționează, cum se face contarea unui document justificativ) poate deveni o muncă pe care puțini sunt dispuși să o facă. Succesul unei echipe mixte formată din contabili și informaticieni vine și din modul în care fiecare poate înțelege abstractizările făcute de celălalt. Aceasta impune găsirea unui limbaj comun în care un obiect (cont, chitanță, balanță) să ajungă să aibă o aceeași semnificație atât pentru contabil cât și pentru informatician.

Încapsularea este un proces prin care se ascund detaliile de implementare a comportamentului astfel încât interfața (adică partea publică a clasei) oferită de clasa de obiecte să fie clară, în concordanță cu elementele obținute în urma abstractizării. Încapsularea este proprie programatorilor și se poate considera că este bine făcută atunci când diagramele de clase stabilite pentru sistemul informatic nu suferă modificări profunde.

Moștenirea se manifestă într-o ierarhie de clase. În acest caz, ierarhia nu trebuie înțeleasă ca o subordonare. Moștenirea din cadrul claselor de obiecte se referă la modul în care clasele de obiecte își partajează proprietățile și comportamentul. Exemplul clasic este cel al mamiferelor. *Mamifere* este clasa care are proprietățile cu valorile: membre:4; tip_sânge: cald și comportamentul dat de: *NaștePuiVii*. O clasă aflată pe un nivel inferior și care moștenește clasa *Mamifere* este *Canide* care moștenește proprietățile și comportamentul clasei *Mamifere* dar care are suplimentar proprietatea *CuloareBlană* iar comportamentul se îmbogățește cu metoda *Latră*. În lumea contabilității o clasă poate fi *Document* care are proprietățile *Număr* și *Data* iar o clasă care moștenește clasa *Document* poate fi clasa *Chitanțe* care are suplimentar proprietățile *Client* și *Suma*. Despre clasele care moștenesc se mai spune că sunt **clase derivate** iar despre clasele moștenite se spune că sunt **superclase** sau clase părinți. Conceptul de moștenire este important în procesul de abstractizare pentru că permite ca părțile comune (care se suprapun) să fie tratate în manieră identică o singură dată în clasa părinte dar fiind valabile în toate clasele derivate, părțile care disting clasa derivată de clasa părinte se tratează doar în clasa derivată fără ca să fie influențată clasa părinte.

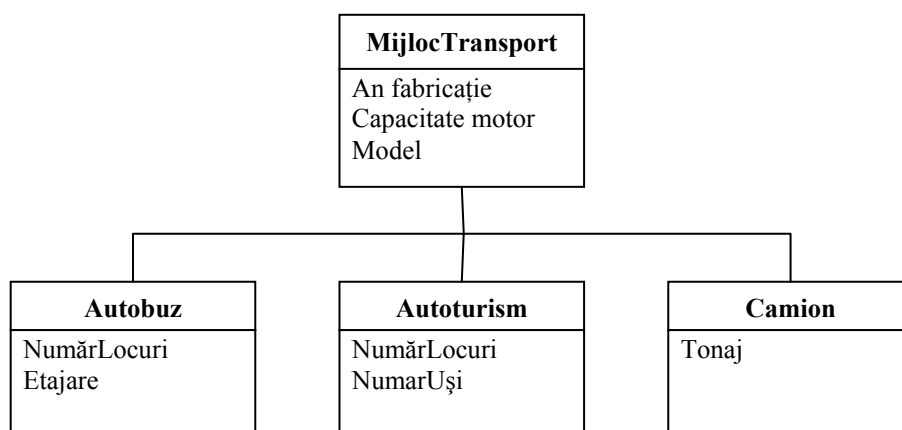


Figura 3.3 Reprezentarea relației de moștenire dintre superclasa *MijlocTransport* și clasele derivate *Autobuz*, *Autoturism*, *Camion*

Polimorfismul reprezintă capacitatea unei metode de a fi funcțională în clase de obiecte distincte. De exemplu, pentru superclasa *Documente* se poate defini metoda *Procesare* care va fi definită în fiecare dintre clase în mod diferit.

Alte concepte utilizate ([DAV03]):

- acțiunea – operațiile instantanee, neîntrerupte, asociate evenimentelor;
- activitatea – operațiile care durează în timp, întreruptibile;
- agregarea – obiectele sunt reprezentate de componente în cadrul unui obiect privit ca întreg;
- asocierea – un ansamblu de legături; se identifică prin nume unic și poate avea atașată o multiplicitate care exprimă numărul de asocieri în contextul dat;
- relația/legătura/asocierea dintre obiecte – o conexiune logică/fizică dintre obiectele aparținând unei clase;

- mesajul – cerere adresată unuia sau mai multor obiecte prin care se pot solicita date sau se modifică starea obiectului (obiectelor);
- starea obiectului – se definește prin valorile proprietăților unui obiect la un moment dat; starea se modifică prin acțiunea unor stimuli externi obiectului (evenimente);
- tranziția – trecerea obiectelor de la o stare la alta prin utilizarea unor mesaje specifice.

Etapele UML

Utilizarea UML presupune parcurgerea următoarelor etape ([LUN03]):

- definirea problemei – se stabilesc caracteristicile principale și modul de funcționare a activității de implementat;
- structurarea soluției – se determină și se detaliază cerințele utilizatorului final;
- analiza sistemului – se analizează cazurile de utilizare și se extrag conceptele cele mai importante cu care lucrează sistemul;
- construirea soluției – se realizează o analiză detaliată a modelului pentru a se obține o variantă care să fie ușor de translatat în cod scris într-unul sau mai multe limbaje de programare;
- proiectarea sistemului care presupune proiectarea de ansamblu (se definesc subsistemele și relațiile dintre acestea) și proiectarea de detaliu (se detaliază subsistemele și se rafinează descrierea relațiilor dintre acestea);
- implementarea sistemului – se efectuează programarea și se construiește diagrama componentelor software rezultate.

Structurarea soluției este etapa care trebuie să elaboreze modelul comunicării dintre echipa de analiză informatică și echipa care reprezintă utilizatorii finali (echipa care are cunoștințe despre funcționarea sistemului informațional contabil). Comunicarea dintre aceste două echipe este foarte importantă pentru că noțiunile folosite se deosebesc funcțional; astfel echipa informaticienilor pot înțelege prin „tabel” un obiect al unei baze de date care are atașată o structură de câmpuri cu atribute limitate din punct de vedere funcțional; un membru al echipei utilizatorilor poate înțelege „tabel” fie ca un document de înțrire care conține un registru scris de mână al încasărilor dintr-o zi, fie un rezultatul unei sortări a denumirilor terților. Documentele elaborate în cadrul acestei etape cuprind diagramele cazurilor de utilizare.

Analiza sistemului este etapa în care se studiază diagramele cazurilor de utilizare (create în cadrul etapei precedente) și se extrag elementele cu care lucrează sistemul. Pentru a se evidenția relațiile dintre elemente, atributele și comportamentul lor, se construiesc următoarele diagrame:

- diagramele claselor;
- diagramele obiectelor;
- diagramele de stare și/sau diagramele de activitate;
- diagramele de secvență;
- diagramele de colaborare.

Câteva dintre aceste diagrame vor fi studiate într-un capitol viitor.

Construirea soluției este etapa în care se încearcă obținerea unui model îmbogățit care să fie mai ușor de translatat într-un limbaj de

programare. Intervenția programatorilor poate determina crearea unor clase noi, relații, diagrame noi care trebuie să reflecte gestionarea datelor (de exemplu în prezența unei baze de date), comunicare cu exteriorul sistemului (de exemplu pentru comunicarea prin e-mail etc.).

Proiectarea de ansamblu definește arhitectura sistemului (prin subsistemele și interacțiunile dintre ele) și, pentru fiecare subsistem, se descriu printr-o *proiectare de detaliu* clasele, se rafinează comportamentele prin adăugarea și/sau modificarea metodelor claselor obținute în etapele anterioare.

Implementarea sistemului reprezintă etapa de programare propriu-zisă. În această etapă se folosesc diagramele create anterior și *diagramele componentelor* software, se scrie codul sursă și se obțin componentele software.

BIBLIOGRAFIE

1. [DAV03] Davidescu, N. D. – *Proiectarea sistemelor informatice prin limbajul Unified Modeling Language*, Editura All Beck, București 2003
2. [LUN03] Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare”, Editura Economică, București, 2003
3. [MIH03] Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, Editura ASE, București 2003

1. <http://www.sparxsystems.com.au>

TESTE DE EVALUARE

1. Ciclul de realizare a sistemului informatic definit de:

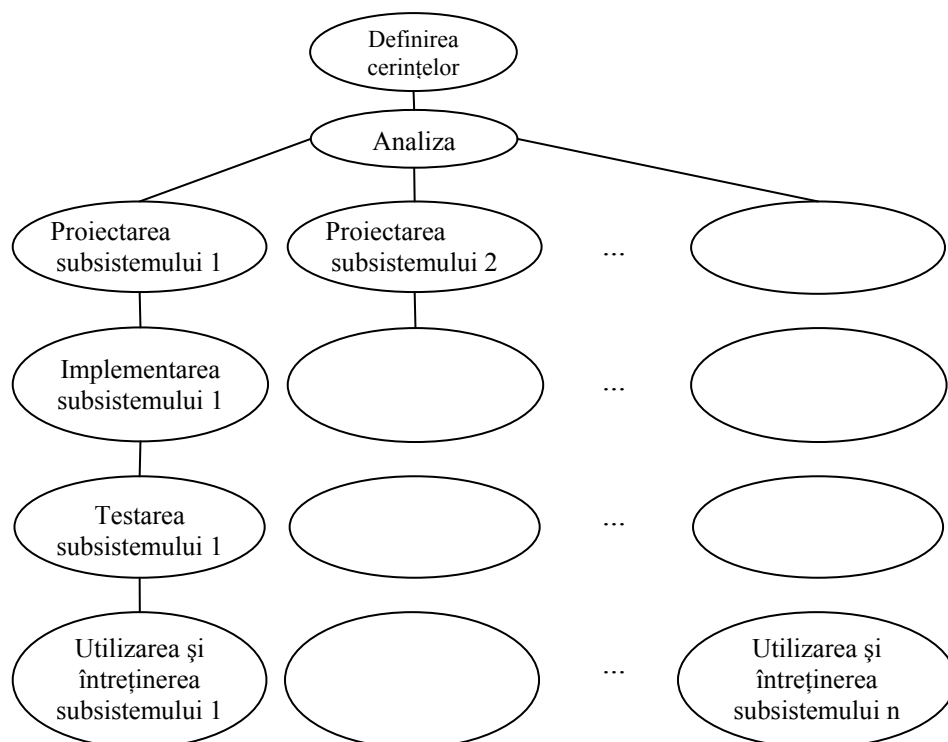
2. Ciclul de viață a sistemului informatic definit de:

- a) intervalul de timp $CV = [T1, T2]$ unde $T1$ – reprezintă momentul în care s-a decis cumpărarea sistemului iar $T2$ reprezintă abandonarea sistemului;
- b) intervalul de timp $CV = [T1, T2]$ unde $T1$ – reprezintă momentul în care s-a decis elaborarea sistemului iar $T2$ reprezintă abandonarea sistemului;

c) intervalul de timp $CV = [T1, T2]$ unde $T1$ – reprezintă momentul în care s-a decis documentarea sistemului iar $T2$ reprezintă casarea sistemului.

3. Etapele realizării sistemelor informatice prin intermediul Unified Modeling Language sunt:

4. Completați schema de mai jos astfel încât să reprezinte un model incremental de realizare a sistemelor informatice:



5. Etapele modelului în cascadă sunt:

6. Clasa grupează:

- a) obiecte cu același nume și număr de identificare;
- b) obiecte cu aceeași structură și stări diferite;
- c) obiecte cu aceeași structură și același comportament.

TEMA 4. MODELAREA SISTEMULUI INFORMATIC DE CONTABILITATE

CONȚINUT

- 4.1. Specificații generale ale metodei Unified Modeling Language
- 4.2. Diagrame utilizate de UML

REZUMAT

Unified Modeling Language (UML) este potrivit pentru modelarea sistemelor informatice de contabilitate datorită unor caracteristici cum ar fi: folosește abstractizări prin care structurile complexe contabile devin accesibile pentru analiștii informaticieni și designeri nespecialiști în contabilitate; utilizează modelarea vizuală și permite dezvoltarea unei ierarhii de modele, vederi și diagrame.

OBIECTIVE

Tema propusă are ca scop înțelegerea primară limbajului UML și a modului în care se poate folosi în modelarea pentru realizarea sistemelor informatice de contabilitate.

4.1 SPECIFICAȚII GENERALE ALE METODEI UNIFIED MODELING LANGUAGE

Analiza și proiectarea sunt două etape importante ale ciclului de viață al unui sistem informatic de contabilitate. Scrierea programelor fără efectuarea acestor etape înseamnă o decizie care, de cele mai multe ori, se dovedește a fi o decizie greșită cu repercusiuni în etapele de implementare, testare și mentenanță (întreținere) – timpul câștigat aparent prin ignorarea analizei și proiectării se răzbună prin detectarea erorilor logice de către utilizatorul final, prin nerespectarea cerințelor de calitate și de funcționare contabilă.

Obținerea unui model se face printr-un proces de modelare prin care se definesc cerințele individuale ale sistemului informatic de contabilitate. Definirea acestor cerințe poate lua forma unor modele de date, modele funcționale, de proces sau organizaționale, fiecare însoțit de o documentație. Rolul jucat de un model pentru realizarea unui sistem informatic este similar rolului jucat de un proiect de casă întocmit **înainte** de a începe construirea casei.

Sistemul modelat prin intermediul UML va fi descris prin următoarele aspecte ([DAV03], pagina 12):

- aspectele organizaționale – specificul activității utilizatorului final, definirea modulelor etc;
- aspectele non-funcționale – amplasament, coordonare, etc.;

- aspectele funcțional – structura statică, structura dinamică (comportamentală), interacțiunile etc.

UML prezintă următoarele caracteristici care îl face potrivit pentru modelarea sistemelor informatice de contabilitate ([DAV03]):

- este un limbaj universal care se poate folosi pentru realizarea sistemelor informatice;
- folosește abstractizări prin care devin accesibile structurile complexe pentru analiștii informaticieni și designeri nespecialiști în domeniul pentru care se modelează sistemul informatic;
- abordează modelarea obiectuală care asigură eficiență prin reutilizarea componentelor care pot fi privite ca un ansamblu de obiecte inter-cooperante și care se pot organiza într-o structură ierarhică ([DAV03]);
- permite dezvoltarea unei ierarhii de modele, vederi și diagrame (vezi figura 4.1);
- utilizează modelarea vizuală.

Vederile

Vederile prezintă, sub formă de succesiune de diagrame, unele aspecte ale sistemului modelat:

- *vederea cazurilor de utilizare* – descrie modul de funcționare a sistemului și se caracterizează prin:
 - folosește cazurile de utilizare și actorii. Actorii pot fi: interni (fac parte din sistem) sau externi (sunt exteriori sistemului – clienți, furnizori, bănci etc);
 - conține diagrame ale cazurilor de utilizare și, opțional, diagrame ale activităților;
 - destinația lor este formată din utilizatorul final al sistemului informatic, designeri, dezvoltatori (analiști, programatori, testori);
- *vederea logică* descrie modul de funcționare al sistemului din două perspective: statică (prin intermediul diagramelor de obiecte, diagramelor de clase) și dinamică (cu ajutorul diagramelor de activitate, diagramelor de stări-tranziții, diagramelor de colaborări); destinația este compusă din designeri și dezvoltatorii sistemului noi;
- *vederea componentelor* – descrie implementarea modulelor și componentele prin detalii cum ar fi: structurile și tipurile de date, resursele care trebuie alocate (memorie internă, hard-disk etc.);
- *vederea concurență* – este o vedere non-funcțională prin care se descrie structura sistemului prin structurare în procese și procesoare (cu scopul alocării eficiente a resurselor); diagramele sunt destinate dezvoltatorilor; diagramele utilizate diagramele dinamice și diagramele de implementare;
- *vederea amplasamentului* se folosește pentru redarea în mod grafic a locurilor în care vor fi amplasate echipamentele utilizate în cadrul sistemului informatic (calculatoare, case de marcat, puncte în care apar tranzacții bancare etc.), se pot folosi

instrumente de reprezentare ale rețelelor de calculatoare (topologii); diagramele folosite sunt diagramele de amplasament.

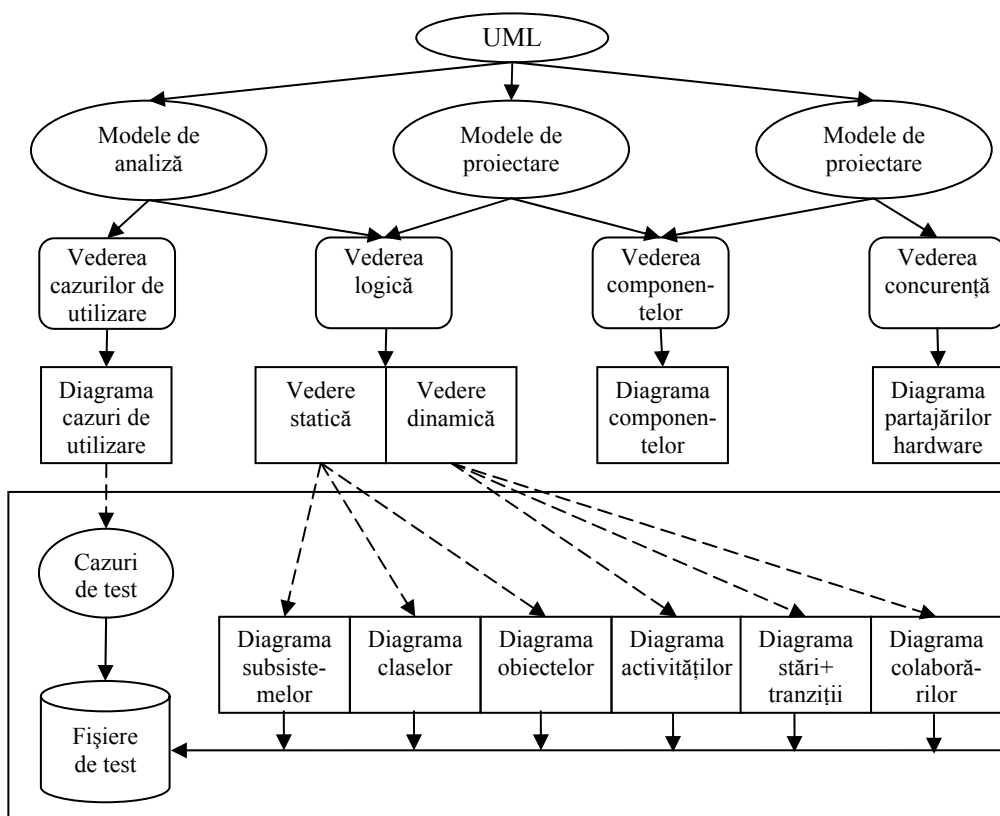


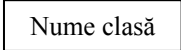
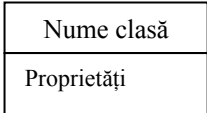
Figura 4.1 Ierarhia de modele, vederi și diagrame utilizate de UML


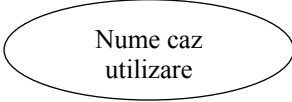
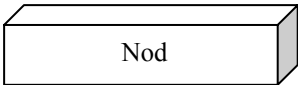
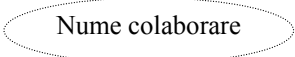
Sursa: Davidescu, N. D. – *Proiectarea sistemelor informatice prin limbajul Unified Modeling Language*, pagina 2

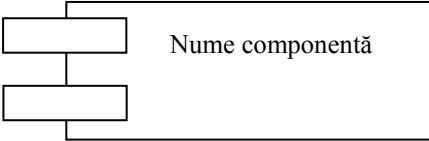
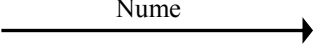

UML pune la dispoziție o extensie pentru modelarea specifică lumii afacerilor ceea ce constituie un avantaj în realizarea sistemelor informatice de contabilitate. Reprezentarea grafică a elementelor în modelele UML este prezentată în tabelul 4.1.

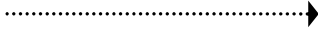
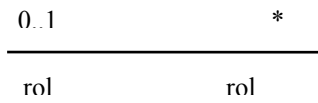
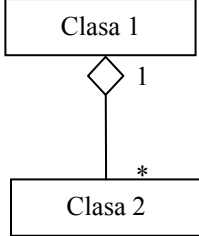
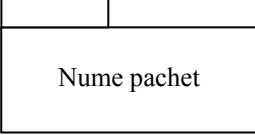
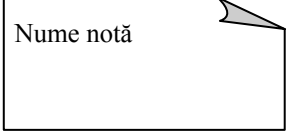
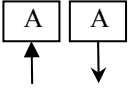
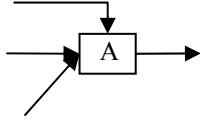
Tabel 4.1

Elementele diagramelor UML

Element	Reprezentare	
(1)	(2)	
Elemente structurale		
Clasă	a) 	b) 

	<p>c)</p> <table border="1"> <tr><td>Nume clasă</td></tr> <tr><td>Proprietăți</td></tr> <tr><td>Operații</td></tr> </table>	Nume clasă	Proprietăți	Operații	<p>d)</p> <table border="1"> <tr><td>Nume clasă</td></tr> <tr><td>Proprietăți</td></tr> <tr><td>Operații</td></tr> <tr><td>Responsabilități</td></tr> </table>	Nume clasă	Proprietăți	Operații	Responsabilități
Nume clasă									
Proprietăți									
Operații									
Nume clasă									
Proprietăți									
Operații									
Responsabilități									
Instanță	<table border="1"> <tr><td>Identificator</td></tr> <tr><td>Proprietate 1: valoare 1 Proprietate 2: valoare 2 ... Proprietate N: valoare N</td></tr> <tr><td>Metoda 1 Metoda 2 ... Metoda M</td></tr> <tr><td>Responsabilitate 1 Responsabilitate 2 Responsabilitate P</td></tr> </table>		Identificator	Proprietate 1: valoare 1 Proprietate 2: valoare 2 ... Proprietate N: valoare N	Metoda 1 Metoda 2 ... Metoda M	Responsabilitate 1 Responsabilitate 2 Responsabilitate P			
Identificator									
Proprietate 1: valoare 1 Proprietate 2: valoare 2 ... Proprietate N: valoare N									
Metoda 1 Metoda 2 ... Metoda M									
Responsabilitate 1 Responsabilitate 2 Responsabilitate P									
Interfață	 Nume interfață								
Caz utilizare	 Nume caz utilizare								
Nod	 Nod								
Colaborare	 Nume colaborare								

(1)	(2)
Componentă	 Nume componentă
Elemente comportamentale	
Interacțiune	 Nume
Stare	 Nume stare
Relații	

Dependență	
Asociere	
Agregare	
Alte elemente	
Pachet	
Notă	
Eticheta	<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>a) </p> </div> <div style="text-align: center;"> <p>b) </p> </div> </div>

Clasa se poate reprezenta în patru variante: a) numai numele clasei; b) numele clasei și proprietăți; c) numele clasei, proprietăți și operațiile care definesc comportamentul (metodele); d) complet: nume, proprietăți, metode și responsabilități. Un tip special de clasă este clasa *activă* care poate iniția control și se reprezintă grafic împreună cu procesele sau firele de execuție.

Componenta este o parte fizică înlocuibilă a unui sistem. *Nodul*, element fizic, poate fi o resursă de calcul, poate deține memorie și capacități de procesare.

Interfața unui obiect se constituie din mulțimea de operații prin care clasa realizează un serviciu. *Colaborarea* reprezintă un ansamblu de elemente care prin comportament cooperativ sunt mai eficiente decât dacă ar fi lucrat separat. *Cazul de utilizare* descrie succesiunea de acțiuni care sunt executate pentru a obține un rezultat de așteptat de către un actor al sistemului. *Interacțiunea* reprezintă mesajele schimbate între obiecte pentru atingerea unui scop.

Etichetele se folosesc atunci când se linia, care reprezintă trecerea de la o activitate la alta: a) este întreruptă sau b) când mai multe linii trebuie unite pentru a reprezenta o joncțiune. Numele etichetelor trebuie ales în așa fel încât să nu se confunde cu nume altor elemente (clase, actori); de obicei

se folosesc fie literele mari ale alfabetului fie cifre sistemului zecimal de numerație.

Un pachet se folosește pentru organizarea elementele în blocuri prin care se simplifică reprezentarea unor diagrame detaliate în altă secțiune a modelului. *Nota* se folosește pentru adăugarea comentariilor referitoare la un element sau un grup de elemente.

O *relație* reprezintă o conexiune între două elemente; după natura acestei conexiuni relațiile sunt:

- de dependență (când modificarea stării unui element determină modificarea stării altui element cu care se află în conexiune);
- de asociere (când fiecare dintre elementele implicate în conexiune sunt independente, fiecare având rolul său în conexiune);
- de agregare (când o clasă conține părți care se pot modela prin alte clase; de exemplu o clasă pentru clasa *Factura* poate fi modelată ca fiind o agregare a claselor *AntetFactura* și *LiniiFactura*.

Cardinalitatea unei relații reprezintă numărul de instanțe care pot fi implicate în relație la un moment dat. Cardinalitate de reprezentă prin limita inferioară și limita superioară, cu observația că dacă nu se cunoaște limita superioară aceasta se indică prin caracterul *; de exemplu, în cazul agregării de mai sus, cardinalitatea pentru clasa *AntetFactura* este 1 iar pentru *LiniiFactura* 1..*.

4.2 DIAGrame UTILIZATE DE UML

Diagramele „sunt prezentări grafice ale unui set de elemente, cel mai adesea exprimate ca un graf de noduri (elemente) și arce (relațiile)”¹. În continuare se vor prezenta diagramele care se pot folosi pentru construirea unui model utilizând UML.

Diagrama de clase

Diagramele de clase se folosesc pentru reprezentarea grafică a claselor și a relațiilor dintre ele. Sunt cele mai folosite diagrame în modelarea sistemelor și ilustrează vederea statică de proiectare a unui sistem care oferă suport în primul rând cerințelor utilizatorilor finali funcționale ale sistemului. Elementele conținute într-o diagramă de clase sunt:

- clasele de obiecte, interfețe și colaborări;
- relații între clase;
- elemente de notare;
- mecanisme de extensie (constrângeri, valori etichetate);
- pachete sau subsisteme;

¹ Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, pagina 5-93

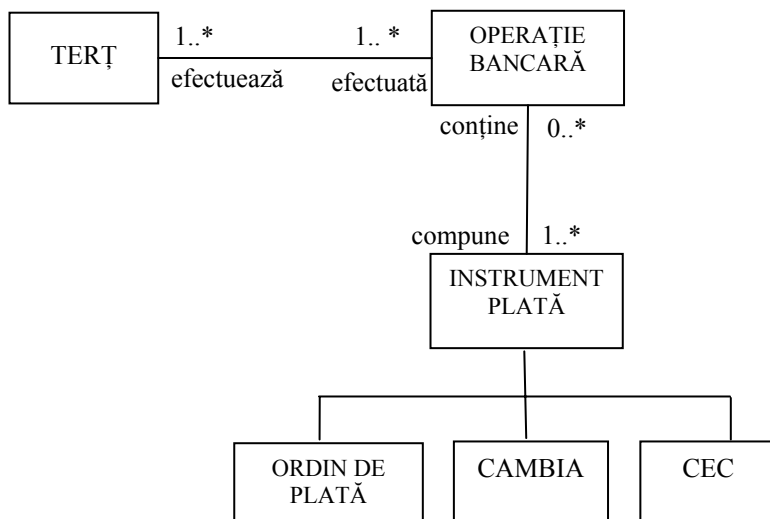


Figura 4.2 Diagrama claselor
Adaptat după: [DAV03], pagina 14

În figura 4.2 diagrama claselor prezintă:

1. clasele implicate într-o operațiune bancară (terț, operație bancară, instrument de plată, ordin de plată, cambia, cec); în această figură, pentru simplificare, nu s-au reprezentat atributele claselor;
2. relațiile dintre clase (reprezentate prin săgeți cu nume, de exemplu *conține*): *Operația bancară* poate conține un document de plată de clasă *ordin de plată*, *cambie* sau *cec*;
3. cardinalitatea relațiilor – în figura 4.2 semnificațiile lor sunt:
 - a) **1..***, pentru clasa *Terț*, înseamnă că unul sau mai multe obiecte de clasă *Terț* poate face mai multe operații bancare;
 - b) **1..***, pentru clasa *Operație bancară*, înseamnă operațiile bancare pot fi efectuate de unul sau mai mulți terți;
 - c) **1..***, pentru clasa *Instrument plată*, înseamnă unul sau mai multe instrumente de plată compun o operație bancară;
 - d) **0..*** înseamnă că un obiect de clasă *Operație bancară* poate folosi zero sau mai multe instrumente de plată de tipurile *ordin de plată*, *cambie* sau *cec*.
4. rolurile claselor sunt:
 - a) *efectuează* pentru clasa *Terț* în relația cu clasa *Operație bancară*;
 - b) *efectuată* pentru clasa *Operație bancară* în relația cu clasa *Terț*;
 - c) *conține* pentru clasa *Operație bancară* în relația cu clasa *Instrument plată*;
 - d) *compune* pentru clasa *Instrument plată* în relația cu clasa *Operație bancară*.

Diagramele obiect

Diagramele obiect reprezintă o variantă a diagramelor claselor și care reflectă instanțele claselor conținând pentru valorile atributelor, relațiile dintre instanțe și, pentru un model rafinat destinat și programatorilor, detalii specifice pentru tipurile atributelor (vezi figurile 4.5 și 4.6).

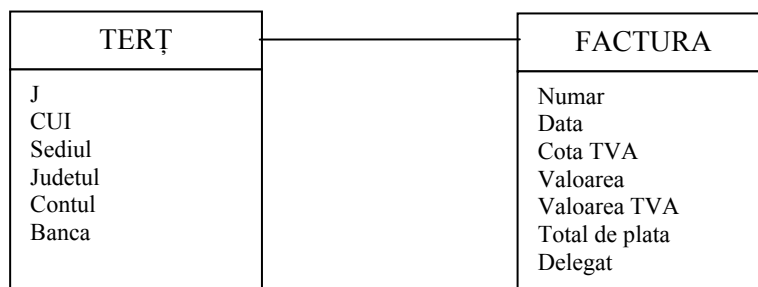


Figura 4.5 Diagrama claselor *Terț* și *Factura*

Figura 4.6 prezintă o instanță a clasei *Terț* și două instanțe ale clasei *Factura*.

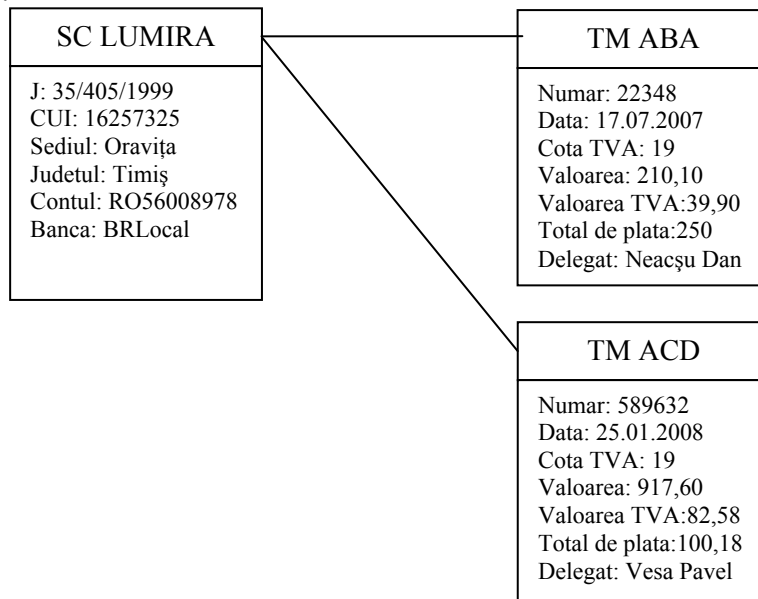


Figura 4.6 Instanțe ale obiectelor *Terț* și *Factura*

Diagramele cazurilor de utilizare

Diagramele cazurilor de utilizare se utilizează pentru modelarea aspectelor dinamice ale sistemului (comportamentului unui sistem, subsistem sau al unei clase) și conține cazuri de utilizare, actori, relații de dependență, de generalizare și de asociere, note și constrângeri și pachete.

Simbolurile folosite sunt prezentate în figura 4.3 (Sursa: [DAV03], pagina 33).

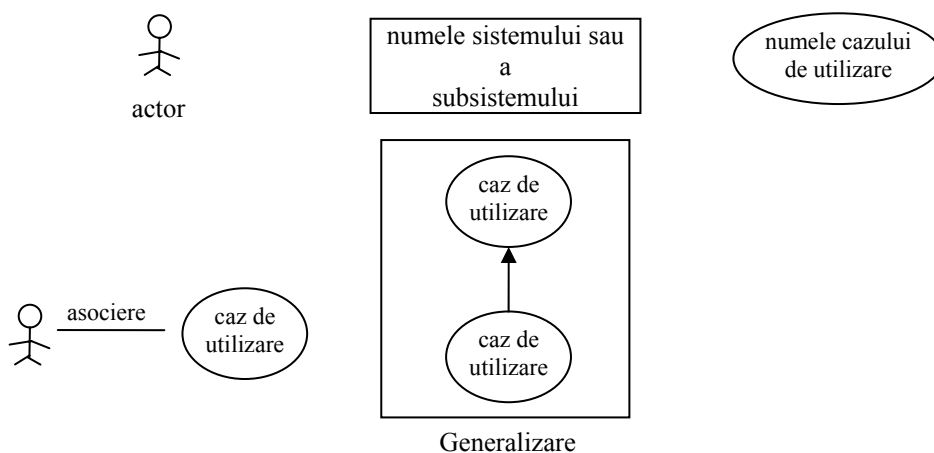


Figura 4.3 Simboluri folosite în diagrama cazurilor de utilizare

Diagramele cazurilor de utilizare se pot completa cu descrieri textuale care conțin informații privind cerințele și funcționalitatea sistemului. Cum actorii (terț este un actor în figura 4.4) se poate reprezenta prin clase, este evident că se pot modela și relațiile dintre aceștia dacă este necesar.

Figura 4.4 prezintă cazul de utilizare pentru eliberarea unei facturi.

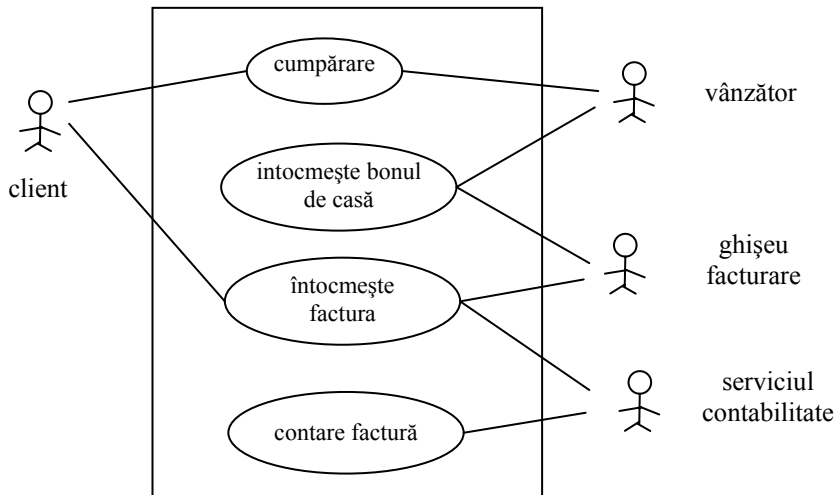


Figura 4.4 Diagrama cazurilor de utilizare în cazul cumpărării unor articole și eliberării unei facturi

Facem precizarea ca dreptunghiul în care sunt încadrate „cumpărare”, „întocmește bonul de casă”, „întocmește factura” și „contare factură” este granița unui subsistem.

Diagramele de stare-tranziție

Diagramele de stare-tranziție completează descrierea obiectelor prin:

- descrierea tuturor stărilor posibile pe care le pot avea obiectele unei clase;
- evidențierea evenimentelor care determină schimbarea stărilor.

Diagramele de stare se întocmesc pentru clasele care au un număr definit de stări.

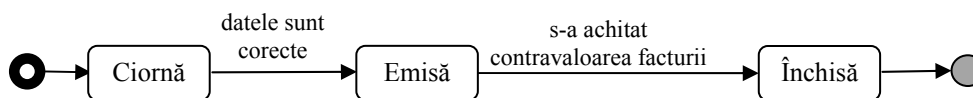


Figura 4.7 Diagrama simplificată a stărilor unei facturi

În figura 4.7, ● reprezintă punctul inițial, iar ● punctul final al diagramei.

Figura 4.8 prezintă o diagramă îmbogățită a stărilor unei facturi.

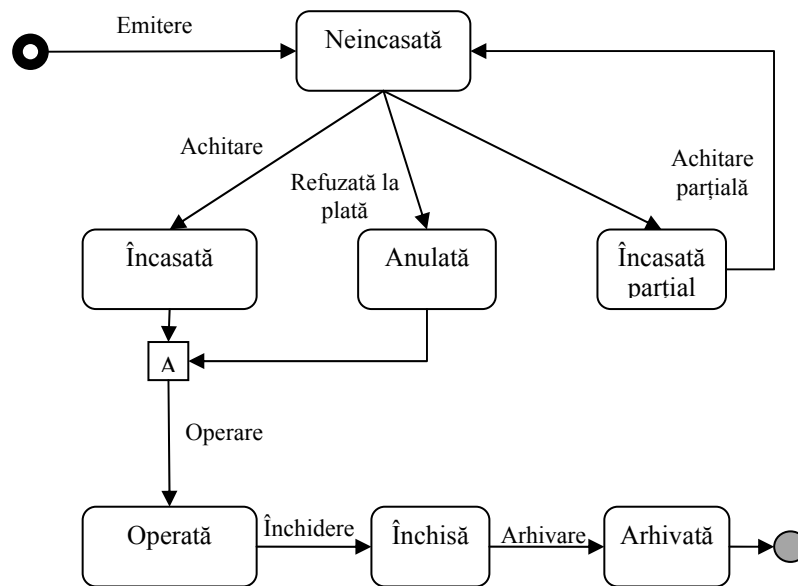


Figura 4.8 Diagrama stărilor unei facturi

Diagramele de stare pot conține informații despre timpul consumat, erori, condiții pentru ca o stare să devină adevărată (cum e „datele sunt corecte” din figura 4.7), expirarea timpilor de așteptare etc. Aceste diagrame sunt utile pentru a se identifica modul în care un obiect își poate modifica starea sub influența unor stimuli. Diagramele de stare pot fi însoțite de tabele în care se pot folosi formule de clacul (de exemplu, pentru a se indica matematic ce înseamnă o factură neachitată).

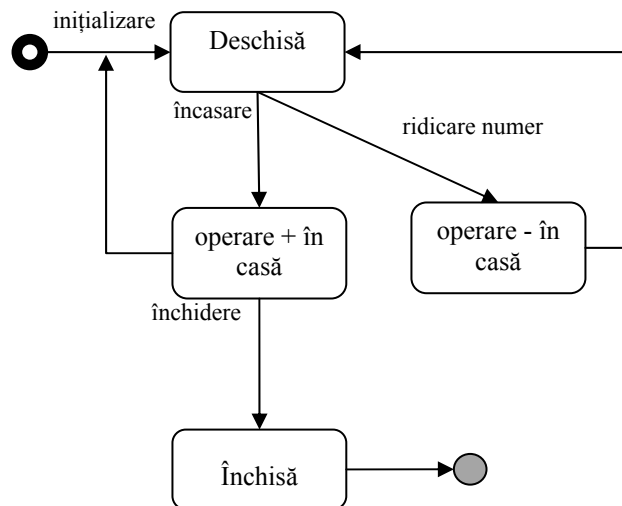


Figura 4.9 Diagrama stărilor unei case de marcat

Stările „operare + în casă” și „operare - în casă” se pot detalia pentru a se scoate în evidență operațiile contabile, documentele emise (de exemplu, chitanța la încasare).

Mai multe diagrame de stări se pot conecta – marcajul grafic este o săgeată întreruptă (vezi figura 4.10).

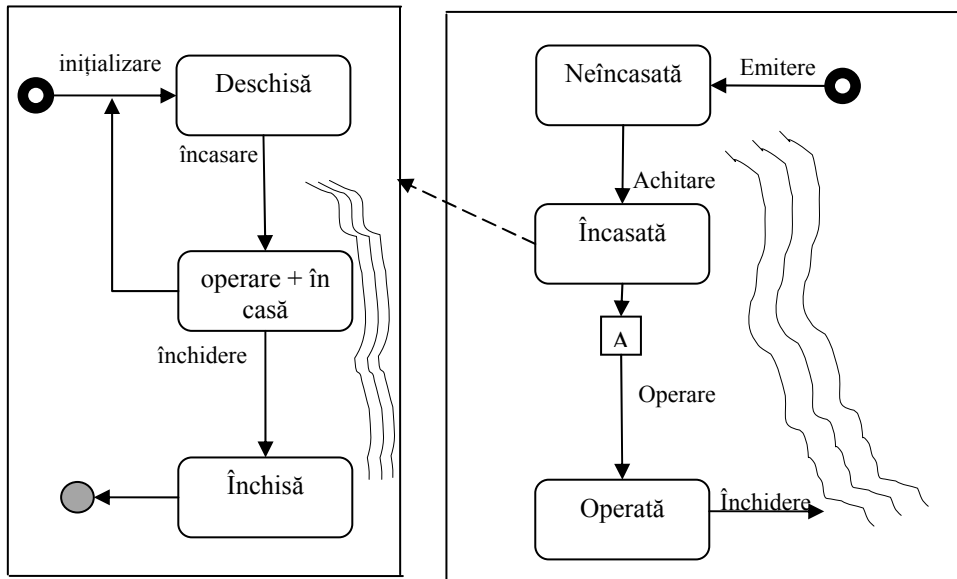




Figura 4.10 Comunicare între subsistemele reprezentate de diagramele din figurile 4.8 și 4.9 în cazul plății unei facturi prin numerar

În figura 4.10 săgeata întreruptă este folosită pentru a indica mesajul transmis din subsistemul diagramei stărilor facturii în subsistemul casei de marcat. Liniile ondulate reprezintă granița dintre partea vizibilă și partea ascunsă a fiecărui subsistem.

Diagramele de activitate

Diagramele de activitate modelează aspectele dinamice ale sistemului informatic și descriu activitățile care se realizează prin operații pentru care se pot prevedea condiții și decizii reflectând astfel și rezultatele aplicării acestora (vezi figura 4.11).

O diagramă de activitate conține ([MIH03], pagina 5-118):

- starea inițială – reprezentată grafic prin ;
- starea finală – reprezentată grafic prin ;
- stări;
- tranziții;
- ramuri – în urma evaluării unei expresii logice, fluxul de control al activității trece de la o activitate la alta în funcție de valoarea de adevăr a expresiei, punctul în care se evaluează o expresie logică se reprezintă grafic printr-un romb;
- bifurcații și îmbinări – apar atunci când există activități care continuă în paralel sau care se îmbină într-un punct; reprezentarea grafică este o bară groasă, verticală sau orizontală;
- culoarele – se folosesc când stările de activitate se pot împărți în grupuri; culoarele se reprezintă prin linii verticale;
- fluxul de obiecte – se folosește atunci când în fluxul de control al activității sunt implicate obiecte pentru care se poate specifica rolul, atributele și starea.

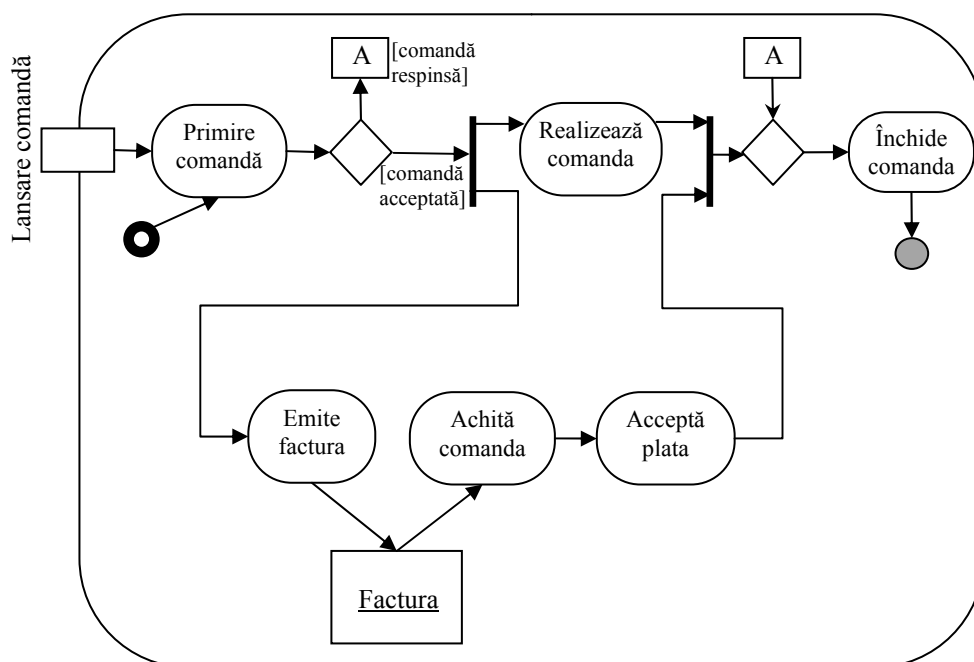


Figura 4.11 Diagrama activităților în cazul unei comenzi

Diagramele de activitate pot fi folosite pentru a reprezenta ([LUN03], pagina 411) acțiunile care se realizează atunci când se execută o operație și activitatea internă a unui obiect.

În figura 4.11 Factura reprezintă o clasă. Dacă se face o detaliere mai amănunțită, clasa se poate completa cu proprietățile ei, metodele, rolul jucat în diagrama prezentată în figură.

Figura 4.12 prezintă diagrama de activitate pentru modulul de raportări manageriale. „Utilizator”, „Modul raportare” și „Gestiune rapoarte” sunt trei culoare ale diagramei cu ajutorul cărora se grupează activitățile. Refuzul cererii de listare se face pentru utilizatorii care nu au dreptul de listare ale unor raporte confidențiale. După ce s-a listat un raport activitate, se poate continua pe una dintre ramurile prevăzute: fie se închide modulul pentru raportare fie se afișează lista rapoartelor disponibile pentru a se emite o nouă cerere de listare. Pentru exemplul din figura 4.12

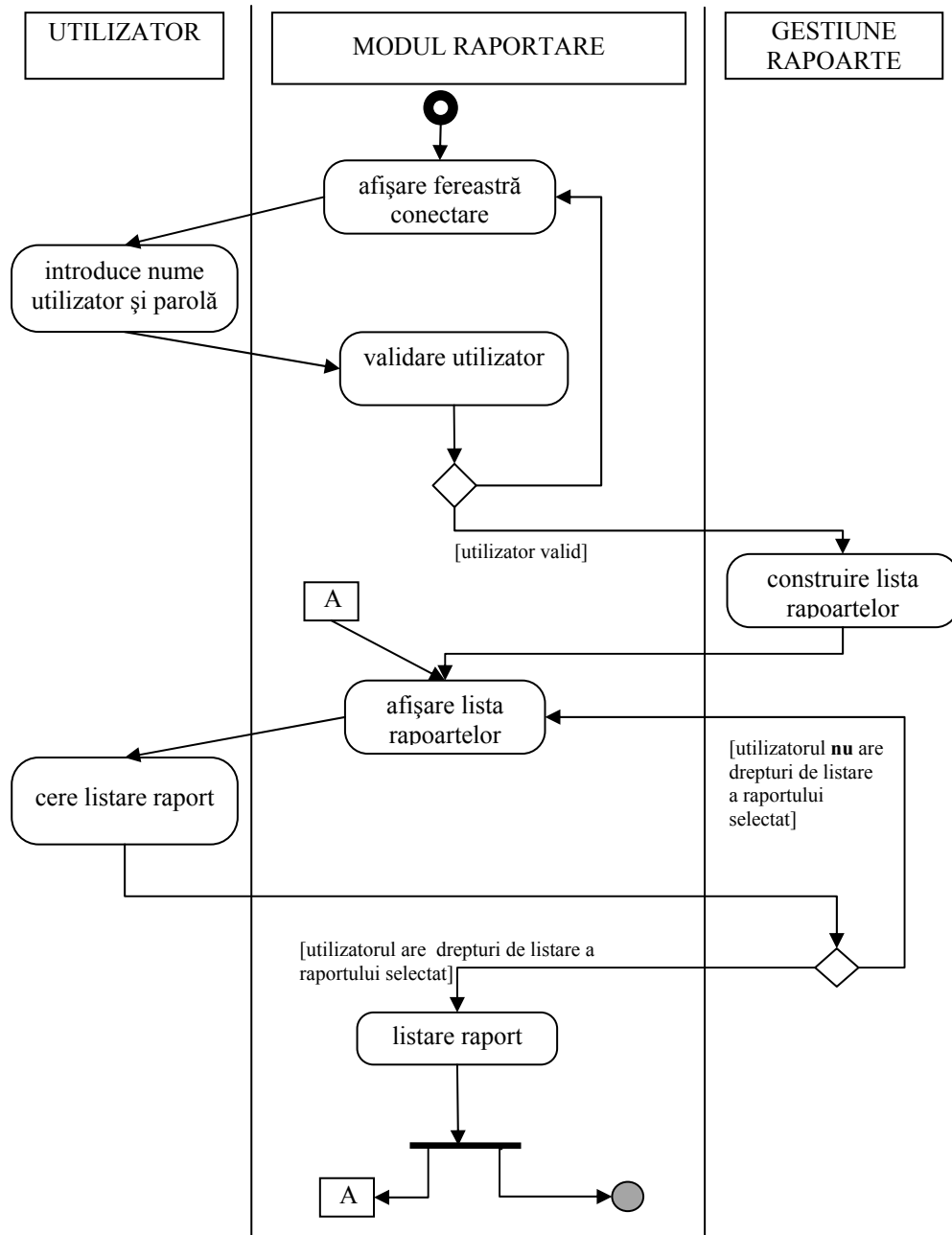


Figura 4.12 Exemplu de diagramă a activităților pentru listarea rapoartelor

Diagramele activităților ajută la identificarea acțiunilor care se pot realiza într-o ordine bine determinată. De asemenea, ele scot în evidență modul în care acțiunile influențează obiectele cu care interacționează.

BIBLIOGRAFIE

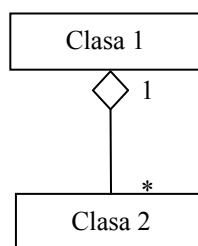
1. [DAV03] Davidescu, N. D. – *Proiectarea sistemelor informatice prin limbajul Unified Modeling Language*, Editua All Beck, București 2003
2. [LUN03] Lungu, I., Sabău, G., Velicanu, M. ș.a. – „Sisteme informatice. Analiză, proiectare și implementare”, Editura Economică, București, 2003
3. [MIH03] Mihalca, Rodica, Fabian, C. – *Realizarea produselor program aplicative*, Editura ASE, București 2003

TESTE DE EVALUARE

1. Precizați cel puțin trei caracteristici ale UML care îl face potrivit pentru modelarea sistemelor informatice de contabilitate:

2. Vederile utilizate de UML sunt:

3. Diagrama din figura de mai jos reprezintă:



- a. două clase independente;
 - b. un element de agregare;
 - c. o asociere condiționată.
4. După natura conexiunii dintre două elemente, relațiile sunt:

5. Precizați diagramele folosite de UML:

-
-
-
6. Întocmiți diagrama stărilor unui aviz de expediție.
 7. Întocmiți diagrama activităților în cazul unei facturi încasate printr-un ordin de plată.
 8. Întocmiți diagrama activităților pentru modulul de introducere a documentelor justificative cu contare automată al unui sistem informatic de contabilitate.
 9. Fie clasa STUDENT cu proprietățile Nume, Prenume, Anul de studiu, Nota și clasa FACULTATE cu proprietățile Denumire, Specializare, An acreditare. Să se întocmească diagrama claselor și să se stabilească rolul și cardinalitatea fiecărei clase.
 10. Pentru clasele de la exercițiul 9, să se întocmească diagrama obiectelor.

TEMA 5. ANALIZA SISTEMULUI INFORMATIC DE CONTABILITATE EXISTENT

CONȚINUT

- 5.1. Obiectivele analizei
- 5.2. Elaborarea modelului fizic al sistemului existent
- 5.3. Elaborarea modelului logic al sistemului existent
- 5.4. Alegerea unui nou sistem informatic de contabilitate

REZUMAT

Analiza sistemului informatic de contabilitate existent este o etapă importantă prin care sistemul devine accesibil în activitatea de reproiectare a sistemului sau de formulare a unor cerințe suplimentare.

OBIECTIVE

Tema propusă are ca scop învățarea modului în care se poate studia un sistem informatic de contabilitate existent.

5.1 OBIECTIVELE ANALIZEI

Analiza sistemului informatic de contabilitate existent este o etapă importantă prin care sistemul devine accesibil în activitatea de reproiectare a sistemului sau de formulare a unor cerințe suplimentare. Analiza sistemului informațional existent este cea mai sensibilă parte a proiectării unui sistem informatic și constă în descrierea sistemului deja existent dar și a celui care urmează a fi introdus. Din cauză că „în economia românească nu există un standard referitor la sistemele informaționale” ([BOB03, pagina 31), analiza sistemului informatic este o problemă spinoasă care iscă divergențe referitoare chiar și la timpul care trebuie alocat acestei etape: un timp prea scurt poate însemna insucces în cunoașterea sistemului, neînțelegerii modului sau înțelegerea precară a modului în care funcționează acesta și nedetectarea locurilor și posibilelor situații în care pot apare erori. Un timp prea lung dedicat acestei etape înseamnă scurtarea timpilor celorlalte etape ceea ce poate duce la rezultate incorecte și/sau soluții incomplete. În ambele situații rezultatul poate fi un produs care nu se ridică la nivelul de așteptare al utilizatorului final.

Cunoașterea mediului în care funcționează sistemul informatic de contabilitate este primul pas care trebuie făcut etapa de analiză. Granița dintre sistem și mediu se poate reprezenta prin *modelul mediului*. Modul în care sistemul interacționează cu mediul se poate reprezenta în diagrama modelului mediului.

Mediul sistemului informatic de contabilitate

Studierea mediului în care va funcționa sau funcționează sistemul informatic de contabilitate începe cu culegerea de informații despre unitatea economică elaborându-se două modele:

- *modelul fizic*: prezintă structura tehnică și operațională, modul de funcționare într-o anumită implementare, cu anumite restricții de ordin tehnic;
- *modelul logic*: prezintă modul de funcționare independent de implementare (acest model împreună cu cerințele utilizatorului este folosit pentru proiectarea sistemului informatic nou).

Studiul mediului intern al unității economice în care funcționează sistemul informatic de contabilitate cuprinde următoarele activități ([LUN03]):

1. *specificare cunoștințelor generale despre organizație* – denumirea, sediul central, filiale, forma juridică, domeniul de activitate, sfera de activitate, oferta de produse, numărul de angajați, punctele de lucru, structura organizatorică (este de folos și alcătuirea unei organigrame), informații despre clienți, furnizori etc.
2. *cunoașterea activităților organizației*, a normativelor și a legislației de reglementare a activității: regulamentul de funcționare, regulamentul de ordine interioară, statutul de funcționare, informații utile despre funcții, relații dintre compartimente, activitățile de interes și detalierea modului de organizare a acestora, resursele folosite etc. Prezentarea activităților se poate face sub formă de text liber, cu organigrame, tabele, imagini sau orice alte elemente prin care se poate descrie mai bine;
3. *prezentarea caracteristicilor de management*: metodele și tehnicile folosite în luare deciziilor, planificarea, organizarea, controlul etc.;
4. *identificarea mijloacelor tehnice* folosite pentru prelucrarea datelor, modul de utilizare, personalul implicat (nivelul de studii necesar, instruirea continuă), performanțele și punctele slabe.

În cazul în care analiza sistemului se face pentru identificarea cauzelor unor deficiențe în funcționare sau pentru proiectarea/reproiectarea unei componente sau a unei părți a sistemului, echipa de analiză trebuie să ia în considerare cerințele utilizatorului și opiniile managerilor care se pot prezenta sub formă tabelară specificându-se informații referitoare la persoana care a emis cerința și descrierea ei (acțiunea atașată, limitele calitative).

Cerințele pot fi clasificate în două grupe mari¹:

- **cerințe funcționale** – se referă la modul în care sistemul nou trebuie să realizeze activitățile: modul în care sistemul interacționează cu alte sisteme, organizarea interfețelor grafice și obținerea rapoartelor, stocarea datelor, accesarea și prelucrarea lor etc.;

¹ Lungu, I., Sabău, G., Velicanu, M. – „Sisteme informatice. Analiză, proiectare și implementare”, Editura Economică, București 2003, paginile 81-82

- **cerințe nefuncționale** – se referă la modul în care resursele trebuie să aducă performanță la nivel funcțional: timpul de acces a datelor, securitatea datelor, arhivarea și refacerea datelor, audit și control, automatizarea unor operații etc.

Structurarea sistemului informațional existent

Unitățile economice sunt sisteme complexe în care se manifestă o multitudine de interacțiuni între elementele constituente și cu mediul exterior. Sistemul se poate descompune după mai multe criterii:

1. *funcțiunile sistemului* – sistemul se descompune din punct de vedere funcțional în subsisteme: financiar-contabil, cercetare-dezvoltare, resurse umane etc.;
2. *activitățile sistemului* – activitățile sunt grupate în funcție de specific (gestiune, salarizare, evidența comenzilor, evidența producției etc.);
3. *organizarea sistemului* – fiecare compartiment sau departament este considerat un element de descompunere: personal, administrație, producție, marketing, contabilitate etc.;
4. *natura componentelor* – sistemul se descompune în funcție de tipul componentelor: materii prime, produse finite, resurse umane etc. și se identifică activitățile asociate cu acestea (producție, desfacere etc.);
5. *conducere* – se identifică a) subsistemele de conducere strategică, operativă și tactică sau b) subsistemul decizional, subsistemul condus și subsistemul informațional.

Pentru orice tip de descompunere, descrierea unui sistem se poate face pe trei nivele de detaliere ale caracteristicilor:

- *generic*: când se folosesc caracteristici generice;
- *parțial*: când se detaliază cel puțin o parte constituantă a sistemului prin specificarea completă a parametrilor care, prin cuantificări, definesc caracteristicile;
- *particular*: când se detaliază toate părțile constituente ale sistemului prin specificarea tuturor parametrilor aferenți caracteristicilor.

Descrierea unui sistem va avea ca finalitate realizarea unor documente care cuprind specificații de proiectare și de implementare.

5.2 ELABORAREA MODELULUI FIZIC AL SISTEMULUI EXISTENT

În timpul elaborării modelului fizic al sistemului informatic de contabilitate existent se vor construi:

- modelul mediului – prin realizarea diagramei de context;
- modelul comportamental – prin realizarea modelului fizic al prelucrărilor și a modelului logic al datelor.

Modelul fizic al prelucrărilor conține descrierea datelor de intrare și a datelor de ieșire, diagramele de flux a documentelor, descrierea entităților externe sistemului și/sau modelului și descrierea proceselor elementare.

Diagramele de flux a documentelor

Diagramele de flux a documentelor, pe scurt DFD, sunt folosite pentru a reprezenta prelucrările dintr-un sistem, parcursul documentelor de la intrarea în sistem până la obținerea rezultatelor, de la emiterea lor până la părăsirea sistemului. Aceste diagrame scot în evidență modul în care documentele sunt distribuite, utilizate, ultimul loc în care sunt folosite, de fapt cam tot ce se întâmplă cu documentele într-un sistem. Diagramele se pot folosi pentru urmărirea și stabilirea controlului intern asupra documentelor, a responsabilităților, slăbiciunilor sau ineficiența comunicării interne.

Pentru a se elabora diagramele fluxului de documente trebuie executați următorii pași preliminari:

1. identificarea părților sistemului implicate în fluxul documentelor;
2. prin chestionarea oamenilor din interiorul unității economice, luarea notițelor manuale detaliate despre fiecare document folosind simboluri distincte pentru fiecare document, tabele, liste, grafuri;
3. transpunerea în format electronic și folosind simboluri standardizate, dacă există;
4. verificarea diagramei luând în considerare următoarele aspecte:
 - a. o diagramă trebuie să aibă un început și un sfârșit în desfășurarea evenimentelor;
 - b. utilizarea comentariilor acolo unde aspectele surprinse nu sunt clare;
 - c. definirea clară a intrărilor, prelucrărilor și ieșirilor;
 - d. documentele să nu fie conectate direct;
 - e. când se utilizează mai multe copii ale unui document, se înscrie numărul de copii;
5. verificarea diagramei din punct de vedere funcțional împreună cu oamenii chestionați anterior și refacerea acelor diagrame care nu corespund;
6. stabilirea datelor de identificare a diagramei prin: numele documentelor, data și creatorul.

Diagramele fluxurilor de documente se pot construi în manieră simplificată (vezi figura 5.1) dar și în manieră detaliată (vezi figura 5.2).

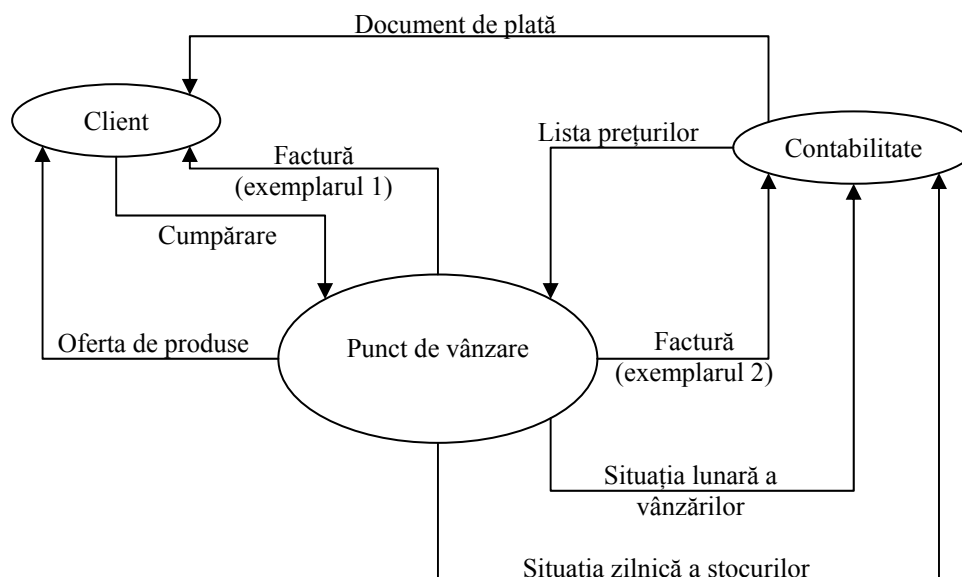


Figura 5.1 Diagramă de flux a documentelor

Diagramele simplificate ale fluxurilor de documente se elaborează rapid din arce direcționale (care pot fi însoțite de explicații scurte care descriu acțiuni și numărul de exemplare ale documentului) și din elipse (în care se încriu numele entităților implicate). Acest tip de diagrame sunt utile pentru a înțelege interacțiunile dintre părțile componente ale sistemului. Figura 5.1 prezintă diagrama de flux simplificată a documentelor în cazul unei unități economice particulare. Această diagramă poate fi mai complexă în cazul unor unități economice mai complexe, de exemplu în cazul unei întreprinderi producătoare pot exista departamente distincte pentru livrări, depozitare și livrare.

Diagramele detaliate de flux a documentelor conțin informații referitoare la locurile și momentele în care apar evenimente sau se desfășoară acțiuni ca:

- documentul apare (este emis sau recepționat) sau se completează pentru prima oară un document tipizat;
- se modifică, prin adăugare sau ștergere, conținutul unui document;
- se manipulează și/sau se deplasează un document – de exemplu este transmis între departamente fără nicio modificare și fără nicio reflectare în contabilitate,
- se execută verificarea cantitativă și/sau calitativă a unui document;
- staționarea temporară a documentului;
- arhivarea sau distrugerea documentului;
- crearea sau generarea unui document din mai multe exemplare.








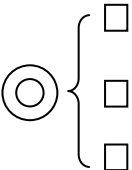




Simbolurile folosite pentru elaborarea diagramelor de flux de documente sun prezentate în tabelul 5.1. Se observă că unele simboluri sunt folosite pentru a specifica două sau mai multe acțiuni care se pot desfășura

simultan, de exemplu simbolul  care se poate folosi a specifica locul în

care se efectuează deodată manipularea, modificarea și verificarea unui document.

Tabel 5.1

Simboluri folosite în elaborarea diagramelor de flux a documentelor¹

Simbol	Utilizare
- 1 -	- 2 -
	Apariția unui document sau completarea pentru prima oară a unui document tipizat
	Modificarea conținutului unui document
	Manipularea unui document
	Verificarea unui document
	Deplasarea documentului
	Staționare temporară
	Arhivare sau distrugere
	Crearea unui document având mai multe copii. Copiile se numerează sau etichetează pentru a se distinge mai ușor în diagramă și în documentația aferentă
	Manipulare și verificare a documentului
	Manipulare, modificare și verificare a documentului
	Linii de influență – generare de documente noi, modificarea conținutului documentelor, manipularea documentelor
	Bloc de simplificare care poate conține oricare din simbolurile de mai sus

¹ Bob, C. A. – *Sisteme informatice în comerț*, Editura ASE, București 2002

Figura 5.2 prezintă diagrama de flux a documentului *factură* în cazul folosirii unui facturier.

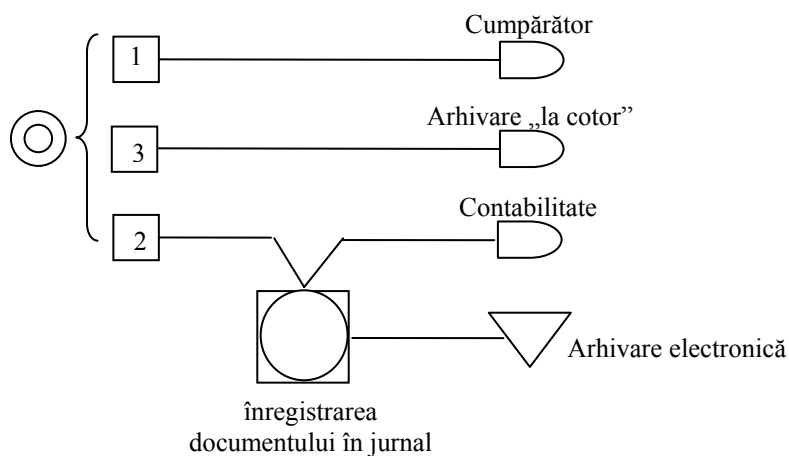


Figura 5.2 Diagrama de flux a unei facturi

Factura se întocmește în trei exemplare care se distribuie astfel: primul exemplar la cumpărător, al doilea este transmis departamentului de contabilitate iar al treilea rămâne la cotor.

Diagramele de flux a datelor

Diagramele de flux a documentelor se pot folosi și pentru elaborarea diagramelor de flux a datelor, pe scurt DFDs. Aceste diagrame se întocmesc respectându-se următoarele reguli ([LUN03]):

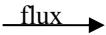
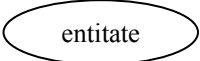
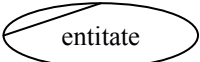
1. pentru a fi identificate mai ușor, procesele și stocurile de date sunt numerotate secvențial;
2. entitățile externe se plasează în jurul stocurilor de date;
3. stocurile de date și entitățile externe se pot reprezenta multiplicat atunci când întretăierea liniilor din graf poate transforma diagrama în una puțin lizibilă.

O diagramă DFDs este constituită, de regulă, din patru elemente de bază (vezi tabelul 5.2):

1. sursele și destinațiile datelor – reprezintă organizații sau persoane care trimit sau recepționează datele utilizate sau recepționate de către sistem;
2. fluxurile de date – reprezentate prin săgeți cu nume mono-direcționate sau bidirecționate;
3. procesele de transformare;
4. stocurile de date.

Tabel 5.2

Simboluri folosite în elaborarea diagramelor de flux a datelor¹

Simbol	Utilizare				
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">nr.</td> <td style="padding: 2px;">localizare</td> </tr> <tr> <td colspan="2" style="text-align: center; padding: 2px;">proces</td> </tr> </table>	nr.	localizare	proces		Procese
nr.	localizare				
proces					
	Flux de date				
	Entitate				
	Entitate duplicat				
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">et.</td> <td style="padding: 2px;">stoc</td> </tr> </table>	et.	stoc	Stoc de date		
et.	stoc				
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">et.</td> <td style="padding: 2px;">stoc</td> </tr> </table>	et.	stoc	Stoc de date duplicat		
et.	stoc				

Tabelul 5.2 prezintă simbolurile folosite pentru întocmirea diagramelor de flux a datelor. *Procesele*, localizate într-un compartiment sau la o persoană, sunt etichetate cu texte care sugerează modul în care se transformă datele.

Un *flux* apare între două procese și este etichetat printr-un substantiv (simplu sau compus) în corelație cu datele sau pachetul de date transmis, de exemplu „stoc final”, „situația vânzărilor”, „încasări” etc. *Entitățile* sunt emițătorii și/sau receptorii de date și pot fi interni sau externi sistemului. După modul de prelucrare a datelor, fluxurile de date sunt de două tipuri:

- de consultare – în acest caz un proces folosește unu sau mai multe stocuri de date (vezi figura 5.3);
- de actualizare – în acest caz un proces modifică datele dintr-un stoc de date (prin operații de adăugare, modificare, ștergere); acest tip de fluxuri sunt reprezentate în unele lucrări de specialitate prin săgeți birecționale (vezi figura 5.4).

Stocurile de date sunt depozite temporare sau permanente de date și pot de trei tipuri, etichetate în funcție de tipul stocului și însoțite de un număr de identificare:

- **M:** manuale – registru, facturier, dosar, arhivă etc.;
- **D:** electronice – pe suport de memorie externă – hard-disk, dischetă, bandă magnetică, CD etc.;
- **T:** temporar, de exemplu un fișier care conține o factură proforma și se stochează temporar până la întocmirea facturii finale.

În figura 5.3 se observă că sunt consultate două stocuri de date pentru a analiza o comandă: stocurile de produse pentru a se verifica dacă

¹ Bob, C. A. – *Sisteme informatice în comerț*, Editura ASE, București 2002

există cantitatea cerută și soldurile pentru a se verifica dacă soldul acoperă plata pentru comandă. Figura 5.4 prezintă cazul actualizării conturilor contabile pe baza unui document.

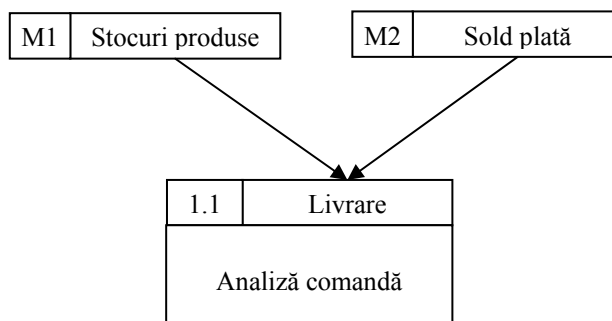


Figura 5.3 Flux de consultare
Sursa: [LUN03] pagina 195

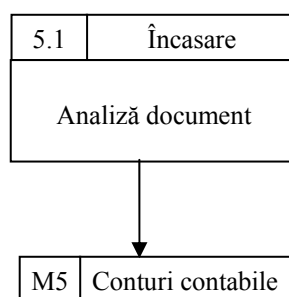


Figura 5.4 Flux de actualizare

Elaborarea diagramelor DFDs se poate face etapizat: în prima fază se vor elabora diagramele contextuale generale care se vor rafina până la nivelul de detaliere cerut. Validarea diagramelor se face de către analiști și de către utilizatori.

Investigarea și reprezentarea datelor

Această activitate este o activitate de importanță mare pentru analiști având un impact mare în etapa de programare. Scopul principal al acestei activități este elaborarea:

- modelului logic al datelor, pe scurt MLD care pune în evidența datele și relațiilor dintre ele în interiorul sistemului;
- catalogul datelor pentru sistemul informatic de contabilitate existent.

Construirea modelului MLD presupune executarea următorilor pași([LUN03], paginile 205-213):

1. **identificarea entităților** – se identifică grupele de date pe care le folosește sistemul, fiecare grup constituindu-se ca o entitate, de exemplu: comandă, produs, beneficiar etc.;
2. **identificarea relațiilor dintre entități** – se construiește **matricea entităților** în care se vor marca prin caracterul * influențele dintre entități (vezi tabelul 5.3), pe baza acestor influențe se construiesc relațiile (asocierile sau legăturile) dintre ele.

Tabel 5.3

Matrice entitate – exemplu

	Beneficiar	Comandă	Linie comandă	Livrare	Linie livrare	Factură	Linie factură	Produs	Intrare	Linie intrare	Furnizor
Beneficiar		*									
Comandă			*	*							
Linie comandă											
Livrare					*	*					
Linie livrare											
Factură							*				
Linie factură											
Produs			*		*		*		*		
Intrare										*	*
Linie intrare											
Furnizor											

Sursă: [LUN03], pagina 206

Tabelul 5.3 a fost construit folosindu-se următoarele informații culese în etapele anterioare:

- un beneficiar poate lansa una sau mai multe comenzi, în acest caz asocierea este 1: N;
- pentru o comandă se execută o singură livrare;
- o comandă poate conține una sau mai multe linii, câte o linie pentru fiecare produs comandat;
- o livrare se încheie printr-o factură; o livrare poate conține una sau mai multe linii, una pentru fiecare produs livrat;
- o factură poate conține una sau mai multe linii, una pentru fiecare produs facturat;
- un produs poate apare în documentele de intrare cât și în documentele de ieșire, mai corect spus în liniile acestora; în acest caz, cu oricare dintre documentele de intrare și/sau ieșire produsul se află în relație M:M (de exemplu o factură poate conține mai multe produse iar un produs poate apare în mai multe facturi);
- o intrare provine de la un furnizor și poate conține una sau mai multe linii, câte una pentru fiecare produs intrat.

3. **elaborarea modelului entitate-asociere** – acest model integrează toate entitățile și relațiile dintre ele determinate în pașii anteriori, relațiile sunt etichetate printr-un verb care exprimă semnificația legăturii și prin cardinalitatea ei (vezi figura 5.5 unde simbolul < reprezintă o cardinalitate mai mare decât unu);

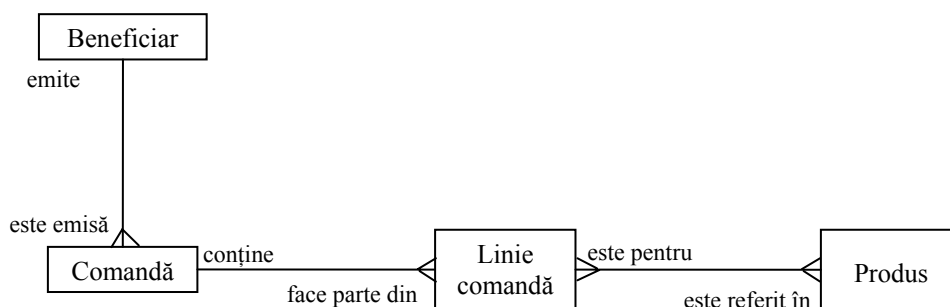


Figura 5.5 Exemplu de model entitate-asociere
Sursă: [LUN03] pagina 209

4. **elaborarea diagramei de corespondență** între stocurile fizice și entitățile logice – fiecare stoc de date este pus în corespondență cu entitățile cu care au fluxuri de actualizare:
- directă – cum ar fi stocul de date pentru produse și entitatea produse, vezi figura 5.6);
 - indirectă și sau multiplă – vezi figura 5.7;

Stocuri fizice	Entități
M1 Stocuri produse	Produs
M2 Sold de plată	Beneficiar

Figura 5.6 Exemple de corespondență directă dintre un stoc fizic și o entitate
Sursă: [LUN03], pagina 209

În figura 5.6, stocurile fizice de date, manuale, pot fi registre tabelate care conțin:

- pentru „Stocuri produse”: denumirea produsului și cantitatea existentă în stoc la un moment dat;
- pentru „Sold de plată”: denumirea beneficiarului și suma care se constituie ca sold de plată.

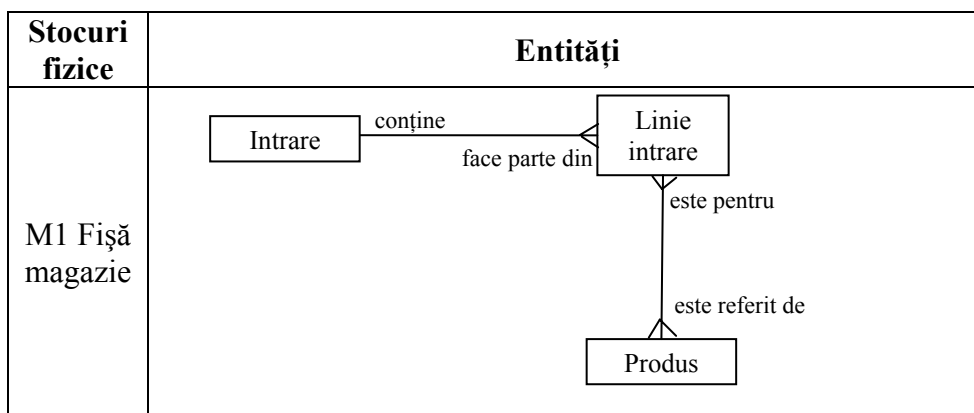


Figura 5.7 Exemplu de corespondență dintre un stoc fizic și mai multe entități
Sursă: [LUN03], pagina 209

În figura 5.7, stocul fizic de date este un stoc manual pentru care se folosesc formulare tipizate. În elaborarea modelului logic, acest stoc de date poate fi dublat de un stoc electronic de date cu aceeași semnificație și utilitate.

5. **descrierea detaliată a entităților**, atributelor acestora și a relațiilor dintre acestea – această descriere se face prin intermediul unor documente standardizate care conțin următoarele informații:
 - pentru descrierea entităților: numele, descrierea, lista atributelor, lista relațiilor și observații;
 - pentru descrierea unui atribut: numele, entitatea care îl conține, tipul entității, descrierea, domeniul de valori pentru care este considerat a fi valid (de exemplu „M”, „F” este domeniul de valori pentru sex), valoarea implicită (de exemplu 0,19 pentru cota TVA), observații suplimentare, și informații necesare pentru etapa de transpunere în limbaj de programare cum ar fi: formatul, și lista utilizatorilor cu drepturile de acces și drepturile de acces, domeniul de grup (care grupează attribute cu semnificații și mod de funcționare asemănător; domeniile de grup se pot descrie detaliat, dacă este necesar);
 - pentru descrierea asocierilor: explicații descriptive; numele entității, tipul de legătură, cardinalitatea și alte observații.
6. **validarea modelului logic al datelor** – se face împreună cu beneficiarul verificând toate aspectele luate în considerare cum ar fi: parcursul datelor, relațiile dintre ele, valorile și domeniile de validare, cardinalitatea.

Crearea catalogului datelor este o activitate complexă care presupune crearea unui dicționar al datelor care conține descrierea atributelor și descrierea domeniilor de grup. Catalogul va deveni un depozit, cu o structură dinamică, de mari dimensiuni folosit în etapa de programare în mai multe scopuri: pentru automatizarea fluxurilor existente în cadrul aplicației, pentru construirea motoarelor de integrare a aplicațiilor și pentru determinarea fluxurilor de date.

5.3 ELABORAREA MODELULUI LOGIC AL SISTEMULUI EXISTENT

Modelul logic al sistemului informatic de contabilitate existent scoate în evidență următoarele aspecte:

- ce face sistemul;
- funcțiile de bază ale sistemului;
- problemele legate de redundanța datelor
- problemele legate de duplicarea proceselor de prelucrare;
- procesele manuale care nu pot fi automatizate complet.

Se obțin diagramele de flux logic pe baza diagramelor DFDs care se descompun în nivele succesive, eliminându-se:

- toate procesele de natură fizică (cum ar fi cele de scriere pe hard-disk care nu interesează în mod direct utilizatorul final acesta presupunând că tehnologia folosită este perfectă adică poate ignora aspectele care țin de scrierea efectivă pe hard-disk);
- stocurile de date care se folosesc ca urmare a constrângerilor din sistem (cele temporare, de sincronizare etc.).

Construirea modelului MLD presupune executarea următorilor pași([LUN03], paginile 214-222):

1. **identificarea stocurilor logice de date** – se realizează prin gruparea datelor înrudite, care se utilizează împreună frecvent sau care se utilizează des în același timp; gruparea trebuie să respecte următoarea regulă: un stoc logic conține una sau mai multe entități, dar o entitate poate să aparțină unui singur stoc logic de date; în mod similar identificării stocurilor fizice de date se stabilesc diagrama de corespondență între stocurile logice de date și entitățile logice și diagrama de corespondență între stocurile logice de date și cele fizice;
2. **înlăturarea dependențelor fizice și temporale** – se elimină din diagramele modelului fizic informațiile următoare: localizarea proceselor, periodicitatea și momentele de timp ale execuției proceselor, caracterizările fizice ale documentelor (de exemplu, faptul că o factură se va tipări pe o coală de dimensiune A4);
3. **derivarea proceselor logice** – acest pas trebuie să elimine redundanțele care există la nivel de procese și să înlocuiască stocurile fizice de date cu stocurile logice de date;
4. **derivarea fluxurilor logice** – acest pas trebuie să stabilească numai fluxurile de informații utilizate efectiv de fiecare proces;
5. **gruparea proceselor elementare și construirea unei ierarhii ale entităților**;
6. **verificarea diagramelor**;
7. **elaborarea documentației** – documentația se compune din toate diagramele fluxurilor de date ale modelului logic.

Odată încheiată această etapă se poate face evaluarea sistemului informatic de contabilitate existent prin evaluarea următoarelor:

1. performanțele și limitările sistemului:
 - a. îndeplinirea obiectivelor, funcțiilor, sarcinilor de bază și de exercitare a conducerii;
 - b. oportunitatea, completitudinea și suficiența informațiilor destinate conducerii;
 - c. timpul de răspuns al sistemului – intervalul de timp din momentul transmiterii unei cereri din partea conducerii până la momentul primirii răspunsului trebuie să fie scurt;
 - d. calitatea și precizia informațiilor obținute;
 - e. calitatea și siguranța fluxurilor informaționale;
 - f. posibilitățile de control;
 - g. timpii optimi privind reacția la apariția unor erori și corecția acestora;

- h. gradul de integrare a sistemului informațional în corelație directă cu gradul de automatizare a prelucrărilor;
- 2. gradul de pregătire a unității economice pentru implementarea sistemului informatic de contabilitate nou:
 - a. existența cunoștințelor și disciplinei tehnologice;
 - b. posibilitățile de instruire și autoinstruire în ceea ce privește utilizarea computerelor și a produselor informatice etc.

Nivelul de pregătire al unei unități economice pentru implementarea unui sistem informatic de contabilitate nou este greu de stabilit pentru că intervin o multitudine de variabile: de la suma limitată în ceea ce privește achiziționarea până la intoleranța personalului în fața schimbării modului de lucru cu care s-au obișnuit.

5.4 ALEGEREA UNUI NOU SISTEM INFORMATIC DE CONTABILITATE

Decizia de schimbare a unui sistem informațional existent poate interveni ca urmare a etapei de analiză. În cazul existenței unui sistem de contabilitate care prezintă deficiențe majore (depășire morală, insecuritate în funcționare) care se pot repara cu costuri mari în timp și bani, conducerea unității economice poate prefera achiziționarea unui sistem de contabilitate nou. Alegerea unui sistem de contabilitate nou se înscrie în categoria problemelor multiatribut sau multicriteriale pentru că este o decizie care trebuie să țină cont de o mulțime de atribute/criterii dintre care unele fiind contradictorii:

- criterii obiective – cum ar fi: prețul, costul abonamentului de întreținere a modificărilor în concordanță cu legislația;
- criterii subiective, intangibile – cum ar fi: ergonomia, interfața prietenoasă;
- incertitudini – cum ar fi: securitatea garantată a datelor.

Problemele multiatribut se modelează sub formă matriceală (vezi tabelul 5.4) unde:

- C_i , $i = 1, 2, \dots, n$ sunt criteriile utilizate în luarea deciziei, se recomandă ca n să nu fie mai mare de 10;
- A_j , $j = 1, 2, \dots, m$ sunt acțiunile posibile, în cazul nostru produsele software de contabilitate analizate;
- a_{ij} , $i = 1, 2, \dots, n$, $j = 1, 2, \dots, m$ sunt consecințele posibile.

Tabel 5.4

Matricea deciziilor pentru problemele multi atribut

$A_j \backslash C_i$	C_1	C_2	...	C_n
A_1	a_{11}	a_{12}	...	a_{1n}
A_2	a_{21}	a_{22}	...	a_{2n}
...
A_m	a_{m1}	a_{m2}	...	a_{mn}

Această matrice se traduce în viața reală astfel: conducerea implicată în luarea unei decizii de achiziționare, va constitui o echipă care formată din m membri care va stabili criteriile care trebuie luate în considerare; de exemplu: operatorul va stabili ergonomia, inginerul de sistem siguranța în funcționare, contabilul șef automatizarea prelucrării contabile a documentelor dar și prețul mic și costurile de întreținere cât mai scăzute ș.a.m.d.

Criteriile neobiective vor primi valori pe o scală subunitară, de exemplu pentru criteriul *interfață* se stabilește următoarea scală:

- 0,75 – interfața este foarte prietenoasă (nu este încărcată, culorile sunt potrivite, butoanele de declanșare a unei operații sunt la îndemână etc.);
- 0,5 – interfața este puțin prietenoasă;
- 0,25 – interfața nu este prietenoasă;
- 0 – interfața este greoaie.

Stabilirea criteriilor este urmată de ierarhizarea lor (stabilirea importanței) folosindu-se o scală de la 1 la p , unde p este numărul persoanelor implicate în luarea deciziei, vezi tabelul 5.5 unde linia K este linia sumei valorilor de ierarhizare și conține indicii de depărtare iar n_{ij} sunt notele de ierarhizare acordate de fiecare persoană fiecărui criteriu. Ierarhizarea criteriilor devine astfel un proces în care este implicată toată echipa.

În funcție de valoarea urmărită, criteriile sunt:

- de minim – atunci când se dorește ca valoarea luată în discuție să fie cât mai mică, de exemplu prețul produsului software;
- de maxim – când se dorește ca valoarea să fie cât mai mare, de exemplu, interfața să fie cât mai prietenoasă.

Pentru fiecare criteriu, se calculează indicele de depărtare

$$K_j = \frac{(N_j)_{\max}}{N_j} \text{ unde } (N_j)_{\max} \text{ reprezintă nota maximă care se poate}$$

acorda înmulțită cu numărul de persoane care acordă note criteriilor.

Tabel 5.5

Ierarhizarea deciziilor

$P_i \backslash C_i$	C_1	C_2	...	C_n
P_1	n_{11}	n_{12}	...	n_{1n}
P_2	n_{21}	n_{22}	...	n_{2n}
...
P_p	n_{p1}	n_{p2}	...	n_{pn}
K	$K_1 = \sum_{i=1}^p n_{i1}$	$K_2 = \sum_{i=1}^p n_{i2}$...	$K_m = \sum_{i=1}^p n_{im}$

Următorul pas este calcularea *matricei de depărtare* – aceste valori sunt subunitare $1-q$ unde q se calculează în funcție de tipul criteriului (de minim sau de maxim) astfel:

- pentru criteriile de minim – raportul se face între valoarea criteriului și cea mai mică valoare dintre toate valorile criteriului;
- de maxim – raportul se face între valoarea criteriului și valoarea cea mai mare dintre toate valorile criteriului.

În final se alcătuiește *matricea de apartenență* folosindu-se linia K din tabelul de ierarhizare a deciziilor, matricea de depărtare prin formula $Z_{ij} = e^{-X_{ij}K_j}$ unde Z_{ij} sunt gradele de apartenență, X_{ij} sunt valori din matricea de depărtare iar K_j sunt coeficienții de depărtare. În matricea de apartenență pe într-o coloană distinctă se însumează valorile pe toate liniile și se alcătuiește clasamentul unde alternativa cu suma cea mai mare ocupă primul loc.

Exemplu: alegerea unui produs software de contabilitate utilizând criteriile multiatribut

Problemă

La firma **LocalAuto SRL** se dorește implementarea unui sistem informatic de contabilitate în condițiile în care contabilitatea s-a efectuat manual.

Rezolvare

Administratorul firmei a format echipa decizională formată din nouă persoane: el însuși, contabilul-șef, doi operatori pe calculator și specialiști ai firmei de consultanță.

În urma analizei sistemului informațional de contabilitate existent, s-au elaborat următoarele cerințe minimale pentru produsul software:

- C1. prețul să fie mai mic de 3500 de lei;
- C2. abonament pentru actualizarea modificărilor în concordanță cu schimbările legislației să permită actualizarea prin intermediul Internetului;
- C3. să existe posibilitatea plății în rate a aplicației și a abonamentului pentru actualizările periodice;
- C4. să se facă instruirea la instalare și o instruire permanentă prin intermediul telefonului și/sau Internetului;
- C5. interfața grafică să fie prietenoasă;
- C6. să existe suport tehnic în zilele lucrătoare până la ore târzii;
- C7. utilizarea aplicației să se poată face pe minim trei calculatoare;
- C8. utilizarea aplicației să se facă sub incidența sistemului de operare Windows XP;
- C9. aplicația să fie modulară și să conțină un modul special pentru contabilitatea financiară.

Pentru criteriul **C3 – plata în rate** – s-a stabilit următoarea scalare:

- 0 – nu există posibilitatea plății în rate;
- 0,25 – plata în rate se face anual;

- 0,5 – plata în rate se face trimestrial;
- 0,75 – plata în rate se face lunar;
- 0,9 plata în rate se poate face printr-o perioadă specificată de client.

Pentru criteriul **C4 – instruire** – s-a stabilit următoarea scalare:

- 0 – nu există posibilitatea instruirii la instalare;
- 0,25 – instruirea se face doar la instalare sau ulterior prin plata unui abonament de instruire;
- 0,5 - nu se face instruire la instalare dar există posibilitatea instruirii permanente;
- 0,75 – instruirea se face la instalare, prin ofertă de documentație și permanent (prin intermediul telefonului și/sau Internetului).

Pentru criteriul **C5 – interfața grafică** – s-a stabilit următoarea scalare:

- 0 – neprietenoasă;
- 0,25 – puțin prietenoasă;
- 0,5 – mediu prietenoasă;
- 0,75 – foarte prietenoasă.

Ofertele luate în considerare sunt următoarele aplicații contabile: **SagaC**. (<http://www.sagasoft.ro>), **ContaSQL** (www.cometa.ro), **EasyCont** (<http://www.sasory.ro>), **CielConta** (<http://www.ciel.ro>)

Tabel 5.6

Exemplu. Matricea deciziilor

$A_j \backslash C_i$	C1 Preț achiziție + prețul abonamentului (criteriu de minim)	C2 Plata în rate (criteriu de maxim)	C3 Instruire (criteriu de maxim)	C4 Interfața (criteriu de maxim, intangibil)
A_1 (Saga)	350	0,25	0,5	0,5
A_2 (ContaSQL)	156	0,75	0,75	0,25
A_3 (EasyCont)	1350	0	0,5	0,75
A_4 (CielConta)	2200	0	0,25	0,25

După studierea ofertelor disponibile s-a constatat că următoarele cerințe sunt respectate de către toate aplicațiile studiate: C6, C7, C8 și C9 așa că nu vor apare în matricea deciziilor, vezi tabelul 5.6.

Determinarea coeficienților de importanță a criteriilor adică **ierarhizarea criteriilor** este prezentată în tabelul 5.7.

Tabel 5.7

Exemplu. Ierarhizarea criteriilor

$P_k \setminus C_j$	C1	C2	C3	C4
P ₁	4	1	4	3
P ₂	4	2	4	3
P ₃	3	4	4	3
P ₄	4	4	4	4
P ₅	4	4	4	2
P ₆	3	4	1	4
P ₇	1	1	2	1
P ₈	2	4	1	1
P ₉	1	4	1	2
K_j	26	28	25	23
(d) $K_j = \frac{(N_j)_{\max}}{N_j} = \frac{36}{N_j}$	1,385	1,286	1,440	1,565

Se calculează matricea de depărtare, prezentată în tabelul 5.8

Tabel 5.8

Exemplu. Matricea de depărtare

$A_i \setminus C_j$	C1	C2	C3	C4
A ₁ (Saga)	0,554	0,667	0,333	0,333
A ₂ (ContabSQL)	0,000	0,000	0,000	0,667
A ₃ (EasyCont)	0,884	1,000	0,333	0,000
A ₄ (CielConta)	0,929	1,000	0,667	0,667

$$\begin{aligned} \bar{X}_{11} &= 1 - \frac{156}{350} = 1 - 0,446 = 0,554 & \bar{X}_{12} &= 1 - \frac{156}{156} = 0 \\ \bar{X}_{13} &= 1 - \frac{156}{1350} = 1 - 0,116 = 0,884 & \bar{X}_{14} &= 1 - \frac{156}{2200} = 1 - 0,071 = 0,929 \\ X_{21} &= 1 - \frac{0,25}{0,75} = \frac{0,75 - 0,25}{0,75} = \frac{0,50}{0,75} = 0,667 \\ X_{23} = X_{24} &= 1 - \frac{0}{0,75} = 1 - 0 = 1 \\ X_{31} &= 1 - \frac{0,5}{0,75} = \frac{0,75 - 0,5}{0,75} = \frac{0,25}{0,75} = 0,333 \end{aligned}$$

Se calculează matricea de apartenență folosind (d) din tabelul 5.7 e^{-x}

Exemplu. Matricea de apartenență

$A_i \setminus C_j$	C1	C2	C3	C4	Suma	Clasament
A ₁ (Saga)	0,464	0,424	0,619	0,593	2,101	III
A ₂ (ContabSQL)	1,000	1,000	1,000	0,352	3,352	I
A ₃ (EasyCont)	0,294	0,276	0,619	1,000	2,189	II
A ₄ (CielConta)	0,276	0,276	0,383	0,352	1,288	IV
K_j	1,385	1,286	1,440	1,565		

În final alternativa candidată cea mai bună este produsul software de contabilitate **ContabSQL**.

BIBLIOGRAFIE

- [LUN03] Lungu, I., Sabău, G., Velicanu, M. ș.a. – *Sisteme informatice. Analiză, proiectare și implementare*, Editura Economică, București, 2003
- [BOB02] Bob, C. A. – *Sisteme informatice în comerț*, Editura ASE, București 2002

- [AUGxx] <http://mis.aug.edu>

TESTE DE EVALUARE

1. Ca urmare a studierii mediului în care funcționează un sistem informatic de contabilitate se vor elabora două modele:

- a. modelul contextual și modelul logic;
- b. modelul resurselor și modelul documentelor;
- c. modelul fizic și modelul logic.

2. Pentru a fi analizat, un sistem se poate descompune după mai multe criterii cum ar fi:

3. Întocmiți diagrama simplificată de flux a documentelor în cazul unui punct de vânzare care eliberează numai bonuri de casă.

4. Întocmiți diagrama detaliată de flux a documentului factură în cazul în care facturile sunt emise prin intermediul unui calculator.

5. Dați câteva exemple de stocuri de date manuale, altele decât cele enumerate mai sus:

6. Pentru construirea modelului MLD în cazul unei unități de învățământ public, determinați entitățile de interes pentru contabilitate.

7. Schițați modelul entitate-asociere în cazul unui aviz de expediție.

8. Pașii elaborării modelului fizic al sistemului de contabilitate existent:

9. Dați câteva exemple care pot fi folosite pentru a stabili gradul de pregătire a unei unități economice pentru implementarea unui sistem informatic de contabilitate nou.

TEMA 6. SECURITATEA ȘI CONTROLUL SISTEMELOR INFORMATICE DE CONTABILITATE

CONȚINUT

- 6.1.** Securitatea și valoarea informației
- 6.2.** Sursele de riscuri
- 6.3.** Auditul sistemelor informatice de contabilitate

REZUMAT

Sistemele informatice de contabilitate funcționează într-un mediu în care conține surse de riscuri care trebuie studiate cu atenție pentru a se lua măsurile de siguranță și control care se impun astfel încât funcționarea sistemului să se realizeze corect.

OBIECTIVE

Tema propusă are ca scop asimilarea unor cunoștințe referitoare la:

- problemele legate de securitatea;
- riscurile asociate mediului sistemului informatic de contabilitate;
- auditul sistemului informatic de contabilitate.

6.1 SECURITATEA ȘI VALOAREA INFORMAȚIEI

Valoarea unui produs software de contabilitate se poate exprima din două puncte de vedere:

- al clientului ca suma maximă de bani pe care un client este dispus să o plătească în schimbul produsului informatic, luând în considerare caracteristicile sale calitative, conjunctura relației cerere-ofertă și prețurile produselor similare ale concurenților;
- al producătorului ca suma minimă a costurilor de producție.

Ca urmare putem spune că valoarea unui sistem informatic de contabilitate este o caracteristică greu de cuantificat. A da valoare contabilă unui întreg sistem informatic este o problemă dificilă chiar și pentru contabili. Deși bunurile intangibile sunt incluse în balanțele contabile, nu există metode exacte de a le da o valoare bunurilor bazate pe tehnologie atât din motive subiective, intangibile, cât și obiective, cum ar fi:

- valoarea nu se suprapune exact peste prețul de achiziției sau costurile de dezvoltare ale unui sistem informatic;
- percepția valorii unui același sistem informatic depinde și diferă de la un utilizator la altul;
- valoarea depinde foarte des de criterii cum ar fi: rapiditatea cu care este obținută, oportunitate și relevanța ei;
- informațiile interne ale unei unități economice nu se pot testa pe piață.

Stabilirea unei valori a întregului sistem informațional devine o preocupare atunci când trebuie înlocuit sau extins. În practică, de obicei, nu se face nici o analiză imediată după implementarea sistemului nou, pentru a se verifica îmbunătățirile aduse de către acesta la nivelul valorii informației.

Există multe activități pentru care securitatea și controlul sunt foarte importante¹, de exemplu:

- serviciile bazate pe răspunsul imediat către consumator (de exemplu, dacă un client al unei bănci face o tranzacție folosind un terminal sau Internetul, va dori să vadă instantaneu modificările din cont chiar dacă tranzacția se face de fapt a doua zi);
- utilizarea unei baze de date centralizată (cum ar fi urmărirea traseului unui colet poștal la nivel național);
- utilizarea sistemelor informatice în procese care pot afecta siguranța și sănătatea populației (de exemplu monitorizarea din centralele nucleare);
- lucrul într-un domeniu în care răspunsul trebuie dat în timp real (de exemplu, monitorizarea bursei de către broker-i);
- monitorizarea și controlul traficului (în aeroporturi, pe străzi etc.).

Afacerile se desfășoară în condiții de risc, dar aceasta nu înseamnă că sunt binevenite alte riscurile noi. Tehnologia informației, implicată în cam toate ariile unei afaceri, și-a demonstrat în timp fragilitatea: a introdus incertitudini noi și riscuri noi, s-a dovedit a fi sensibilă la erori, incidente, fraude și alte tipuri de atacuri cu impact negativ nu numai asupra activității unei unități economice dar și, de multe ori, asupra succesului în afaceri a acesteia. Și totuși, chiar și în aceste condiții de nesiguranță, afacerile au devenit tot mai dependente de tehnologia informației.

Securitatea calculatoarelor a fost gândită inițial pentru păstrarea informațiilor secrete din domeniul militar. În lumea afacerilor securitatea calculatoarelor a pătruns ulterior, după ce afaceriștii au ajuns la concluzia că nu doreau ca, nici întâmplător nici cu intenție, propriile sisteme să fie „deschise” spre exterior și nici ca datele să fie distruse sau alterate din interior.

Modelul de securitate militar (vezi figura 6.1) este unul construit pe patru nivele și răspunde unor exigențe mari de securitate. Pornind de la acest model, într-un sistem informațional mai puțin rigid se pot implementa diverse alte modele pe mai multe nivele, cu diverse grade de securitate, în funcție de natura, dimensiunea și rigoarea stabilită de către conducere.

¹ Hawker, A. – „*Security and Control in Information Systems: A Guide for Business and accounting*”, pagina 17

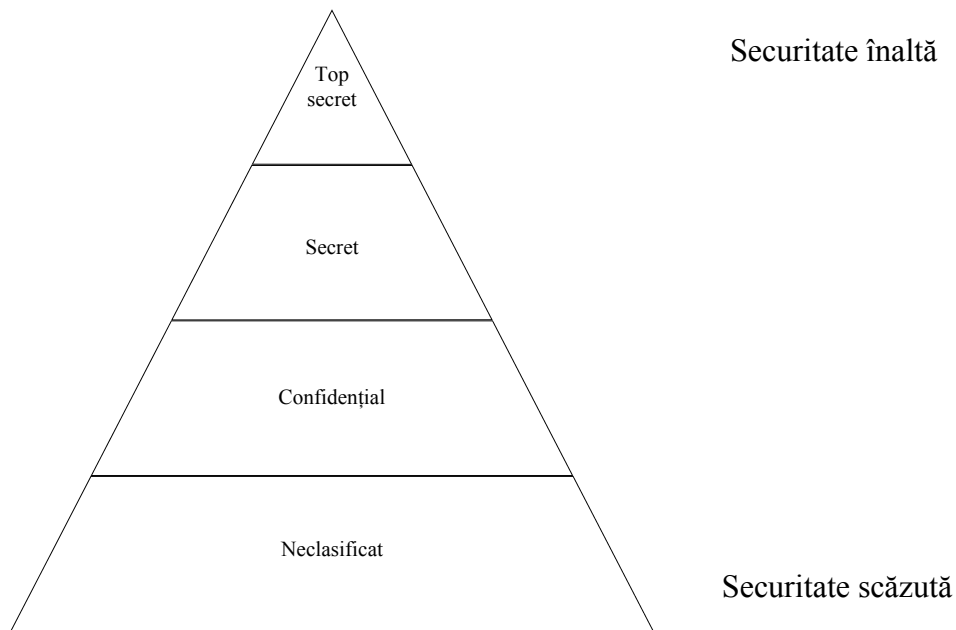


Figura 6.1 Modelul militar al securității.
Sursa: [HAW00], pagina 4

„În mediul economic este foarte important ca modificarea datelor să nu se facă în mod neautorizat și nici ca datele să ajungă la persoanele și/sau companiile nepotrivite”¹ (de exemplu, la posibiii concurenți). Urmărirea securității sistemului informatic de contabilitate (a posibilelor fraude, a integrității și acurateței datelor) se poate face în două moduri:

- prin jurnalul operațiunilor: fiecare operație se salvează într-un jurnal al operațiilor care nu permite decât adăugarea unor date de genul: utilizator, data calendaristică, ora, operația și toate datele despre operația efectuată (de exemplu: suma, nume și număr al actului, beneficiar). Acest jurnal se poate folosi în situații diverse cum ar fi pierderea completă sau parțială a datelor. Jurnalul poate fi util și pentru nivelul de conducere care poate obține informații non-financiare cum ar fi: urmărirea exactă a activității fiecărui contabil, calcularea timpilor alocați operațiilor etc.;
- prin separarea sarcinilor: sarcinile se distribuie distinct contabililor care se „specializează” în lucrul cu anumite operații, cum ar fi încasările.

Obiectivele securității și controlului sistemelor informatice enumerate de Andrew Hawker² sunt: protejarea secretelor, acuratețea datelor, prevenirea falsificării, păstrarea „dovezilor” despre operator, respingerea atacurilor, păstrarea cronologică a accesării autentificate, asigurarea „supraviețuirii” datelor, maximizarea posibilităților de audit.

Se presupune că toate unitățile economice urmăresc să aibă îndeplinite obiectivele securității și controlului. Trebuie făcută observația că

¹ Hawker, A. „*Security and Control in Information Systems: A Guide for Business and accounting*”, pagina 4

² Andrew Hawker „*Security and Control in Information Systems: A Guide for Business and accounting*”, pagina 5

atingerea acestor obiective este foarte importantă pentru organizațiile care manipulează date cu caracter personal, date care trebuie să rămână confidențiale pentru mediul extern al organizației.

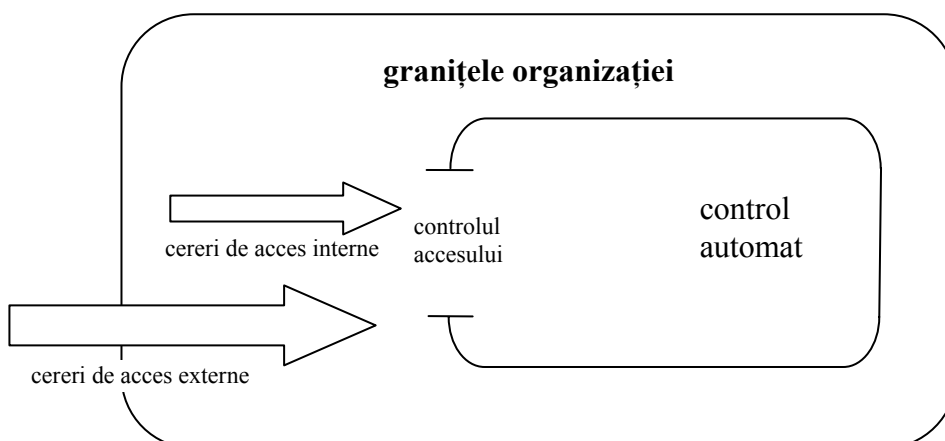


Figura 6.2 Autorizarea accesului într-un sistem informatic
Sursa: [HAW00], pagina 6

Implementarea unui *model de securitate* în cadrul unui sistem informatic de contabilitate presupune identificarea locului/locurilor în care controalele se pot aplica în mod automat sau parțial automat. Aceasta implică stabilirea unor granițe virtuale în jurul unor componente și activități ale sistemului informatic. Ideal este ca toate sistemele informatice ale unei unități economice să se găsească în interiorul acestor granițe. În practică s-ar putea să existe și în afara granițelor așa că se impune verificarea acestora atunci când se face accesarea zonelor cu control intern automat (vezi figura 6.2). Ca urmare a verificării se poate obține autorizația de acces în sistemul informatic de contabilitate.

În interiorul sistemului se vor aplica procedee de control în funcție de modul de urmărire a securității (prin jurnal sau prin separarea sarcinilor). Delimitarea prezentată în figura 6.2 asigură faptul că se poate determina de unde provine o amenințare, adică din interiorul sau exteriorul organizației.

Importanța controlului și protejării informației pornește din faptul că informația are valoare. Într-un mediu concurențial, dacă informația este de importanță mare pentru rivali, informația devine una care trebuie protejată și i se va aplica un nivel de securitate înaltă. Dacă informația este de valoare mică, cum ar fi copia unei chitanțe eliberată pentru o persoană fizică, i se va aplica un nivel de securitate scăzută.

6.2 SURSE DE RISCURI

Toate unitățile economice trebuie să facă evaluarea completă a riscurilor și să implementeze controale interne adecvate pentru a putea stabili programe de management al riscului.

Tipurile și severitatea amenințărilor cresc odată cu dependența afacerilor de sistemele informatice. Aceasta se întâmplă din motive cum ar fi:

- nivelul de operare – multe sisteme informatice funcționează la nivel național sau internațional. Astfel dacă sistemul informatic

al unei bănci devine neoperațional, s-ar putea să apară probleme la nivel național;

- viteza – viteza de lucru și cea de transmitere au crescut astfel că fișiere mari pot fi distruse, copiate sau transmise aproape instantaneu;
- inovația tehnică – tehnologiile noi modifică regulile de bază dar multă lume nu le înțelege atât de bine încât să le folosească în siguranță, putându-se efectua operații despre care nu se știe că sunt riscante. Pe de altă parte, bunii cunoscători ai tehnologiilor informaționale își pot folosi talentele pentru a produce daune fie direct la locul de muncă fie prin pătrunderea din exterior (de exemplu, prin intermediul unei rețele);
- cauze ascunse – de multe ori e greu de descoperit cauza care a stat la baza producerii unei daune (de exemplu efectuarea unei tranzacții bancare duble în condițiile plății unei sume cu ajutorul unui card, blocarea unui card într-un terminal chiar dacă s-au introdus corect datele de identificare).

În continuare vom trata câteva dintre sursele de riscuri.

Sistemele de operare

Sistemele de operare sunt necesare pentru a face toate componentele sistemului de calcul să funcționeze corect și eficient. De obicei un calculator se cumpără cu sistemul de operare gata instalat și care deja are facilități sub formă de programe utilitare, de asistare și de mentenanță a echipamentelor hardware. O atenție deosebită trebuie acordată modului în care se realizează protecția contra accesului neautorizat. Implicit sistemele de operare și aplicațiile pun la dispoziția utilizatorului drepturi depline de acces; aceste drepturi trebuie modificate conform necesităților de securitate ale utilizatorului. Un alt aspect care reclamă atenție e acela al utilizatorilor „uitați” adică un nume de utilizator cu o parolă care se pot folosi de către producător sau de către persoana care a pus în funcțiune sistemul. Riscul este de accesare neautorizată cu drepturi de același nivel ca și ale unui administrator intern al unității economice, pe care beneficiarii sistemului informatic de contabilitate nici nu îl iau în considerare în cazul unei auditări a sistemului.

Sistemele de gestiune a bazelor de date

Sistemele de gestiune a bazelor de date, pe scurt SGBD, se compun dintr-o mulțime de programe care se folosesc pentru definirea, interogarea, protejarea și manipularea unui volum mare de date. Bazele de date trebuie să fie protejate contra amenințărilor intenționate sau neintenționate, „securitatea bazelor de date se referă la elemente de hardware și software, persoane și date”¹. Connolly consideră că securitatea bazelor de date trebuie asigurată corespunzător pentru a preveni situații ca:

¹ Connolly, T., Begg, Carolyn, Strachan, Anne – *Baze de date. Proiectare. Implementare. Gestionare*, pagina 508

- furtul și fraudă (frauda poate apare ca urmare a introducerii intenționate de date eronate, de modificare a documentelor justificative etc.);
- pierderea confidențialității sau pierderea caracterului privat – este foarte importantă, păstrarea secretului despre date, mai ales despre acelea care interesează concurența;
- pierderea integrității sau pierderea disponibilității – pierderea integrității datelor are ca rezultat apariția unor date care nu corespund documentelor justificative; pierderea disponibilității se referă la faptul că datele devin inaccesibile (fie bazele de date s-au corupt din varii motive, fie a avut loc un eveniment hardware).

Daunele pot fi tangibile (cum ar fi deteriorarea unei componente hardware) dar și intangibile (cum ar fi pierderea încrederii unui terț ca urmare a furtului datelor).

Produsele software

Produsele software sunt folosite pentru îndeplinirea funcțiilor afacerilor. Multe dintre acestea (și care pot fi cumpărate pe loc cu o funcționare completă) au fost concepute pentru a îndeplini sarcini generale. Dintre acestea amintim editoarele de documente (Microsoft Word, WordPerfect), programele de calcul tabelar (de exemplu Microsoft Excel, Lotus 1-2-3), și de baze de date (Microsoft Access, SQL Server, Oracle). Alte aplicații au fost create pentru a îndeplini funcții specifice în domenii variate (transferuri bancare online, aplicații de design pe computer pentru asistare în proiectare etc.). Contabilitatea se poate ajuta atât de programe dedicate numai contabilității cât și de programe integrate într-un sistem complex numit ERP¹. Un sistem ERP este o soluție software ale cărei elemente sunt integrate într-o platforma comună. Sistemele ERP actuale realizează integrarea tuturor funcțiilor de conducere ale unei unități economice, (pornind de la planificare, la realizarea gestiunii financiar-contabile, a resurselor umane și terminând cu gestiunea relațiilor clienții și partenerii de afaceri). Un sistem ERP permite, prin simulare a activităților și prin caracterul flexibil și dinamic al aplicațiilor, să se realizeze previziuni, analize calitative și integrarea cu tehnologiile noi de genul e-business și e-comunicare. Exemple de sisteme ERP: *Senior.ERP Suite*, *mySAP ERP*, *B-Org*.

Fiecare din aceste aplicații poate sau nu să aibă elemente de verificare concepute pentru a împiedica accesările neautorizate. Pentru o evaluare a unor verificări competente ale acestor aplicații este necesară dobândirea unor cunoștințe detaliate ale caracteristicilor de verificare a fiecărei aplicații folosite în mod curent într-o unitate economică.

Alte surse de riscuri

Multe sisteme informatice au prevăzute mecanisme de control care sunt proporționale cu gradele riscurilor asociate cu funcțiile îndeplinite de către sisteme. De exemplu, tranzacțiilor financiare li se asociază un grad de

¹ Enterprise Resource Planning – sistemele de planificare a resurselor unității economice

risc mare, un mecanism slab de control poate avea ca urmări furtul datelor celor implicați în tranzacție, alterarea datelor tranzacțiilor și altele.

Virusii, în toate formele lor, sunt un risc care apare în situații ca:

- un angajat lucrează cu o dischetă pe care o folosește și afara unității economice;
- deschiderea e-mail-urilor cu atașamente;
- vizitarea unor pagini de Internet și acceptarea execuției unor componente software (script, fișiere executabile, ActiveX etc.) despre originea căruia nu există date care se pot verifica și care pot avea un caracter distructiv și/sau de culegere a datelor confidențiale.

Tabel 6.1

Riscurile asociate unor acțiuni

Acțiune \ Risc	Furtul și fraudă	Pierderea confidențialității	Pierderea caracterului privat	Pierderea integrității	Pierderea disponibilității
Utilizarea mijloacelor de acces ale unei alte persoane	*	*	*		
Modificarea, copierea, ștergerea neautorizată a datelor	*			*	
Alterarea programelor	*			*	*
Politicile și procedurile necorespunzătoare care permit ieșiri confidențiale pentru un nivel de securitate înalt	*	*	*		
Interceptarea convorbirilor	*	*	*		
Accesul neautorizat sau ilegal	*	*	*		
Crearea unei breșe în sistem	*	*	*		
Furtul de date, programe și echipament	*	*	*		*
Permiterea unui acces prea larg	*	*	*		
Conflictele de muncă				*	*
Pregătirea necorespunzătoare a personalului		*	*	*	*
Vizualizarea și divulgarea neautorizată a datelor	*	*	*		
Alterarea datelor datorită întreruperilor de energie sau supratensiunii				*	*
Calamități				*	*
Introducerea de virusi		*		*	*
Conectarea la Internet	*	*	*		

Criminalitatea informatică a cunoscut o creștere spectaculoasă în ultimii ani. Criminalitatea informatică face parte din crima organizată pentru că: s-a extins la nivel internațional, activitățile ilicite se pot controla de la

distanță (prin intermediul Internetului) și grupările sunt bine structurate și organizate. Persoanele implicate în criminalitatea informatică se desemnează ca fiind „infractori informatici” – sunt persoane care nu trebuie să aibă cunoștințe solide de informatică, pot fi în slujba celor care au resurse pentru construirea echipamentelor „ajutătoare” (bancomatele false). Infractorii informatici folosesc ceea ce este mai nou în domeniu (sisteme, posibilități, modalități de plată speciale) pentru a obține date personale cum ar fi nume de utilizator și parole, numere de cont bancar, numere de carduri, coduri PIN etc. Fraudele cu cărțile de credit cresc ca pondere în criminalitatea informațională. O posibilă explicație este aceea că banii se obțin mai repede și mai ușor decât din alte tipuri de activități ale crimei organizate. Nu este nevoie ca infractorul informatic să între în posesia fizică a cardului. Prin compromiterea bancomatelor (prin folosirea camerelor de luat vederi, fețelor false de bancomat, tastaturi false, dispozitive pentru fanta cardului etc.) se fură informația despre card și se scriu așa numitele blank-uri care se folosesc apoi ca și când ar fi cardurile originale.

Conectarea la Internet, pe lângă beneficii, a însemnat și expunerea în fața unor riscuri ce țin de criminalitatea informatică. Furtul informațiilor despre clienții unui magazin online, este o primejdie la care se expun toți vânzătorii și clienții care folosesc Internetul pentru tranzacții.

Tabelul 6.1 prezintă câteva dintre riscurile asociate unor acțiuni ([CON01, paginile 510-511) care pot avea loc pentru sursele de riscuri descrise mai sus.

6.3 AUDITUL SISTEMELOR INFORMATICE DE CONTABILITATE

Auditul este partea contabilității în care tehnologia informației își găsește o aplicabilitate deplină. Rezultatele financiare tradiționale ale auditului au devenit o industrie matură și se bazează pe legislația de profil și pe standarde elaborate la nivel global (ISA¹), cum ar fi: ISA 401 „Auditul într-un mediu cu sisteme informatice” (Auditing in a Computer Information Systems Environment), ISA 1008 „Evaluarea riscurilor și controlul intern – caracteristici și considerente ale sistemelor informatice” (Risk Assessments and Internal Control – CIS Characteristics and Considerations), ISA 1009 – “Tehnici de audit asistate de calculator” (Computer-Assisted Audit Techniques).

Standardele stabilesc modul în care trebuie să se facă operații ca preluarea și prelucrarea datelor, înregistrarea în conturi. Înregistrarea modificărilor ce se produc în bilanț ca urmare a tranzacțiilor încheiate de societate. Se stabilește și verificarea următoarelor aspecte:

- absența documentelor de intrare, justificative;
- absența probelor materiale de derulare a tranzacțiilor;
- absența posibilităților de accesare și/sau vizualizare a rezultatelor prelucrării.

Obiectivele generale și procesul de audit al situațiilor financiare nu diferă structural de etapele și procedurile comune. Excepțiile apar când

¹ International Standard on Auditing, <http://www.ifac.org/iaasb/>

auditorul dorește cunoașterea programelor de contabilitate, înțelegerea profundă a funcționării acestora pas cu pas, precum și a modului în care acestea răspund cerințelor utilizatorului.

Intrările de bază pentru contabilitate sunt tranzacțiile măsurate în unități monetare. O *urmă-audit* a tranzacțiilor contabile păstrată într-un sistem al unității economice permite utilizatorilor informației să urmărească fluxul datei de-a lungul sistemului. Figura 6.3 este un exemplu de o asemenea urmă care prezintă în paralel un ciclu contabil al unității economice care începe cu datele tranzacției reflectate din documentele de intrare justificative și se termină cu producerea, ca ieșire, a extraselor de cont sau al altor rezultate financiare. Contabilitatea preia datele relevante de intrare din documentele justificative și arhivează documentele pentru o utilizare ulterioară în scopuri de control și auto-control (de exemplu, verificarea cursului valutar pentru o anumită intrare în jurnal).

Un sistem informatic contabil care are o urmă-audit bună permite, de exemplu, unui manager să urmărească datele oricărui document justificativ, prin prelucrare până la locul în care s-a obținut raportul de ieșire. De asemenea sistemul poate să permită unui contabil urmărirea datelor financiare pornind de la balanțele contabile înapoi spre documentele de intrare originale care au determinat tranzacțiile care au influențat balanțele. Ca exemplu, o factură de intrare trebuie să poată fi urmărită prin intermediul urmei-audit de la conturile clientului până la contul debitor și contul creditor. Similar, un contabil poate verifica balanțele pentru conturile creditoare și debitoare prin examinarea tranzacțiilor și a documentelor de intrare originale. Printr-o urmă-audit dezvoltată eficient, un contabil poate urmări datele de-a lungul întregului sistem; această urmărire fiind posibilă dacă oamenii dintr-un sistem pot înțelege pe de-a întregul metodele și procedurile pentru acumularea și prelucrarea datelor. Un rezultat este că se poate reconstrui de către contabili modul în care sistemul manevrează datele. Un sistem computerizat bine proiectat poate îmbunătăți urma-audit prin furnizarea unei liste, a mulțimii tranzacțiilor și a balanțelor conturilor înainte și după ce tranzacțiile au modificat conturile. Pentru unitățile economice care vor să-și dezvolte un sistem de control intern eficient, urmele-audit sunt elemente importante ale acestui control.

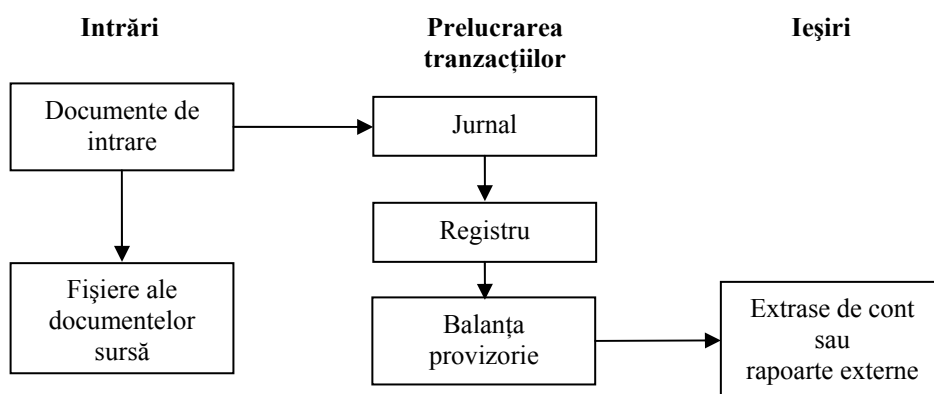


Figura 6.3 Exemplu de urmă-audit financiară
Sursa: [MOS03], pagina 10

Auditorii interni trebuie să examineze componentele unui sistem informatic (hardware, software), mentenanța acestora și alte caracteristici

prin care să se poată determina care dintre asemenea costuri s-au înregistrat corespunzător în bilanțele contabile. De exemplu, componentele hardware și cele software trebuie capitalizate și amortizate în perioade de timp care depășesc cu mult durata de funcționare, perioada de viață în care sunt utile funcționării sistemului informatic de contabilitate iar costurile preplătite pentru întreținere pot fi clasificate ca bunuri și cheltuite numai în perioada pentru care s-au făcut plățile.

Dacă o unitate economică este mică și are numai unul sau doi auditori, aceste persoane trebuie să aibă cunoștințe despre contabilitate, finanțe și altele. Acest tip de auditor este unul care auditează tratarea contabilă a costurilor asociate cu calculatoarele și nu va fi un auditor specializat în auditul sistemelor informatice ([CHA03], pagina 126). Dacă unitatea economică este mare și are un departament de audit intern, auditul se poate face de către unul sau mai mulți auditori cu studii în diverse domenii. În acest caz, pentru un audit profund, se vor examina întregul proces, auditul costului echipamentelor hardware și/sau software fiind doar o parte a auditului. Oricare ar fi modul de control intern, costurile asociate cu echipamentele hardware și cu cele software trebuie să se conformeze normelor legislative.

Investitorii și creditorii care folosesc rezultatele financiare se pot folosi și de alte surse decât auditul pentru informații care să îi ajute în luarea deciziilor. Aceasta se întâmplă din cauza că rezultatele financiare de audit deseori nu devin disponibile într-un timp oportun.

BIBLIOGRAFIE

1. [CHA03] Champlain, J. – *Auditing information systems*, John Wiley & Sons, 2003
2. [CON01] Connolly, T., Begg, Carolyn, Strachan, Anne – *Baze de date. Proiectare. Implementare. Gestionare*, Editura Teora, București 2001
3. [HAW00] Hawker, A. „*Security and Control in Information Systems: A Guide for Business and accounting*”, Routledge 2000
4. [ION07] Ionescu, Gh. – *Nore de curs pentru uzul cursanților de la școala doctorală*, Timișoara 2007
5. [MOS03] Moscové, S.A., Simkin, M. G., Bagranoff, Nancy A. – „*Core Concepts of Accounting Information Systems*”, John Wiley & Sons Ltd, 2003

1. <http://www.ifac.org/iaasb>

TESTE DE EVALUARE

1. Valoarea unui produs software de contabilitate se poate exprima din punctul de vedere al clientului ca:

2. Specificați câteva aspecte care nu se pot cuantifica pentru a da o valoare exactă bunurilor bazate pe tehnologie:

3. Securitatea și controlul sunt importante pentru activități ca:

4. Obiectivele securității și controlului sistemelor informatice sunt:

5. Specificați câteva surse de riscuri:

6. Specificați caracteristicile urmei-audit într-un sistem informatic de contabilitate:
