

Conceptul logic de tranzacție aplicat într-un SGBD pentru automatizarea proceselor de gestiune a stocurilor și de urmărire a comenzilor

Ing. Olga MARINOIU
T&T Group, București

Introducerea noțiunii "tranzacție", în contextul procesului de urmărire a comenzilor, permite evidențierea momentană, dar și a istoricului interacțiunilor pe care acesta le desemnează, furnizând o informare comodă, amplă și corectă a operatorului pe tot parcursul desfășurării.

Cuvinte cheie: sistem software, bază de date, stoc, comandă, tranzacție, ODBC.

În activitatea productivă, în cea comercială sau în cea de servicii, un rol deosebit de important este deținut de procesul de urmărire a comenzilor de la clienți. Acest proces s-a dovedit a fi unul costisitor din punctul de vedere al volumului de muncă și timp solicitat.

Indiferent de natura activității studiate, urmărirea comenzilor se axează în esență pe aceleași aspecte, și anume:

- stabilirea consumului solicitat de o anumită comandă, prin consum înțelegând resurse materiale, umane, financiare și de timp;
- cunoașterea cât mai exactă a stadiului pentru oricare parametru al comenzii.

Diferențe apar în modul efectiv de evaluare a obiectivelor amintite, pentru fiecare tip de activitate economică și, prin urmare, rezultă și particularități de implementare realizate prin intermediul sistemelor de programe. Autoarea acestui articol a studiat și realizat un astfel de sistem software, dedicat activităților comerciale, în cele ce urmează fiind prezentate câteva aspecte rezultate pe parcursul dezvoltării pachetului de programe.

Conceptul logic care stă la baza implementării sistemului de gestiune a stocurilor și de urmărire a comenzilor îl constituie noțiunea *tranzacție*, definită ca ansamblul stărilor, documentelor și al interacțiunilor asociate unei anumite comenzi sau, mai general, unui aspect care se dorește a fi urmărit la un moment dat.

În cazul sistemului de față, s-au delimitat următoarele tranzacții: *tranzacția client*, *tranzacția furnizor* și *tranzacția factură*.

Dar, pentru a putea comenta fiecare din aceste tranzacții, să vedem mai întâi care sunt documentele implicate în gestiunea stocurilor.

Problematika stocurilor este legată de tratarea intrărilor, a ieșirilor și a transferurilor de materiale, cu observația importantă că toate aceste "mișcări" de materiale sunt asociate, de cele mai multe ori, unei tranzacții.

Mișcările (intrări, ieșiri sau transferuri) care s-au stabilit la nivelul întregului sistem și care acționează asupra stocului sunt exprimate prin următoarele tipuri de documente:

- documente de intrare în gestiune:
 - notă de intrare - recepție (NIR);
 - notă de retur (NR).
- documente de ieșire din gestiune:
 - aviz de expediție (AE);
 - bon de consum (BC);
 - notă de casare (NC).
- documente de transfer între gestiuni:
 - notă de transfer (NT).

Din motive de claritate și de eficiență în operare, noțiunea fizică *stoc* a fost separată în două entități logice distincte, și anume: cea de *stoc nerezervat* și cea de *stoc rezervat*, astfel încât suma celor două să furnizeze mereu stocul fizic curent.

În figura 1 este prezentată schematic interacțiunea mișcări-stocuri, la nivelul celor două entități logice. Se poate observa că, între cele două stocuri, nerezervat și rezervat, circulația se face prin intermediul altor tipuri de documente: notă de rezervare (NRez) și notă de repartiție (NRep).

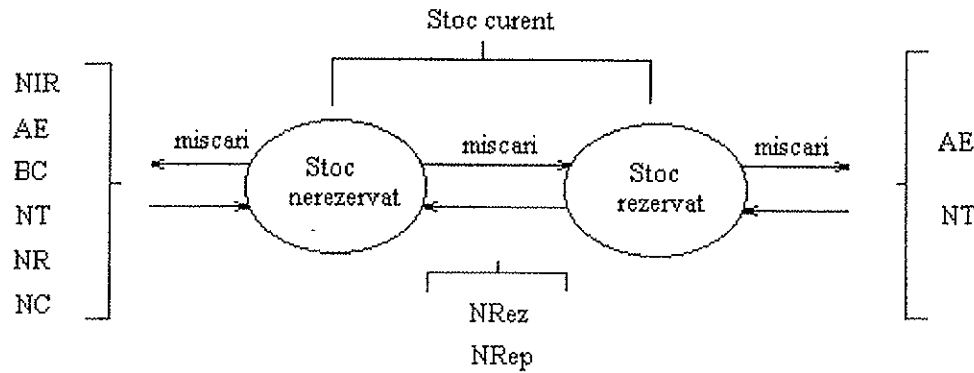


Fig. 1. Interacțiunea mișcări-stocuri

Tranzacțiile client

Tranzacțiile client desemnează ansamblul interacțiunilor client-unitate, cu acces la o stare momentană și păstrând întreaga istorie a acestora. În figura 2 este prezentată sche-

matic noțiunea tranzacție client, putându-se observa elementele sale principale: comenzile de la client; rezervările de materiale pentru client; livrările de materiale către client; evidența facturărilor și a încasărilor în relația unitate-client.

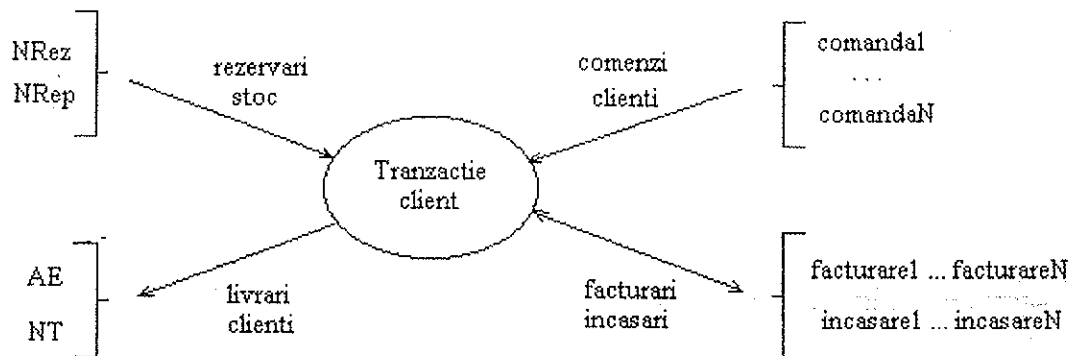


Fig. 2. Tranzacția client

Tranzacțiile furnizor

Tranzacțiile furnizor desemnează ansamblul interacțiunilor unitate-furnizor, cu acces la o stare momentană și păstrând întreaga istorie a acestora. Figura 3 prezintă schematic noțiunea tranzacție furnizor, surprinzând forma-

rea unei astfel de tranzacții, din mai multe comenzi de la clienți (notate generic tranzacții client, întrucât mai multe comenzi de la un client pot intra într-o unică tranzacție client) și lansarea comenzii globale către furnizor.

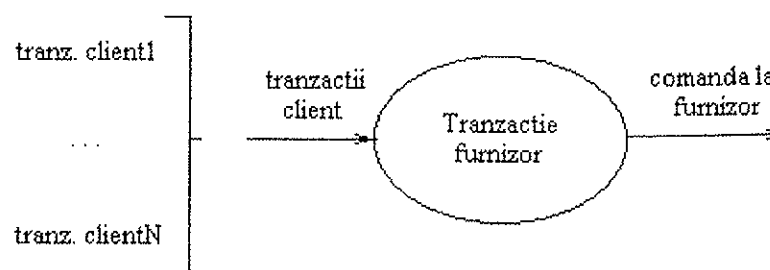


Fig. 3. Tranzacția furnizor

Tranzacțiile factură

O tranzacție factură desemnează ansamblul interacțiunilor unitate-factură de la furnizor, cu acces la o stare momentană și păstrând întreaga istorie a acestora.

Schematic, noțiunea tranzacție factură este prezentată în figura 4. Se poate observa din cadrul figurii că intrările într-o astfel de

tranzacție sunt reprezentate de factura propriu-zisă de la furnizor, care este corectată de beneficiar la momentul recepției mărfurilor, prin evidențierea plusurilor sau a lipsurilor, dar și a rebuturilor față de conținutul facturii.

Pe baza tuturor intrărilor corectate se generează automat intrările propriu-zise, cantitative, în stocul nerezervat.

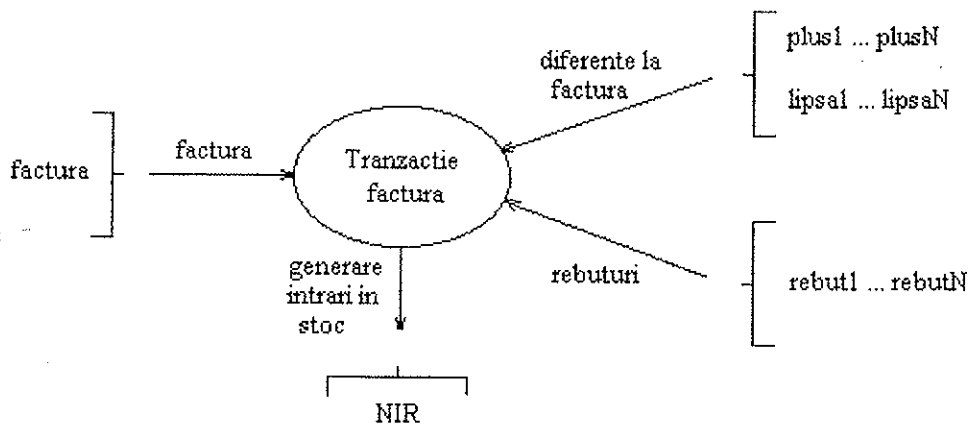


Fig. 4. Tranzacția factură

Concluzii

1. Legat de problematica stocurilor, s-a dovedit a fi utilă și practică separarea și urmărirea stocului curent în două componente: stoc rezervat și stoc nerezervat. În acest mod informațiile de stoc solicitate în mod uzual de utilizator sunt extrase și prelucrate cu ușurință.
2. Introducerea noțiunii *tranzacție* asociată relațiilor societate comercială-client, societate comercială-furnizor, care înglobează ansamblul documentelor și stărilor care justifică derularea unor interacțiuni la aceste niveluri, s-a dovedit a fi foarte eficientă din punctul de vedere al accesului informațional și al eleganței în utilizare.
3. Datorită complexității modulelor de urmărire a acestor interacțiuni, s-au menținut "la cald" (orice mișcare se reflectă imediat și în tranzacție) doar tranzacțiile client, restul tranzacțiilor fiind actualizate "la rece" (orice mișcare nu se reflectă imediat și în tranzacție, ci doar la cererea operatorului).
4. Visual Basic este o soluție convenabilă, ca sistem *front-end*, la dezvoltarea interfeței cu utilizatorul, permițând obținerea unor re-

zultate grafice plăcute și comode în utilizare, având în vedere necesarul de timp solicitat și, nu în ultimul rând, costul acestuia.

5. De asemenea, Visual Basic prezintă avantajul înglobării conceptului de Open DataBase Connectivity (ODBC), deschizând practic posibilitatea de conectare cu orice motor de baze de date, care este prevăzut însă cu facilități de ODBC.

6. În funcție de dimensiunile bazelor de date, de gradul de fiabilitate solicitat acestora în funcționare, precum și de posibilitatea beneficiarului de a investi într-un motor de baze de date mai puternic, ca sistem *back-end*, se poate comuta rapid, fără modificări în programul sursă, dacă acesta a înglobat, la rândul său, facilitățile de ODBC. Referitor la bazele de date, alegerea unui motor de baze de date de tip relațional a permis folosirea cu succes a metodelor de lucru de tip SQL, care s-au dovedit a fi foarte eficiente din punctul de vedere al timpului de acces la informații și al eleganței în scrierea codului de program. În plus, având avantajul de a fi o interfață industrială standardizată de baze de date, SQL permite accesarea și manipularea unei game largi de

produse de baze de date, aspect deosebit dacă se are în vedere protocolul de ODBC.

Bibliografie

1. Boldur-Lătescu, Gh. ș.a. *Analiza sistemelor complexe*, Editura Științifică și Enciclopedică, București, 1982.
2. Bonin, Ch. *Memento SQL, Bases Relationnelles*, Eyrolles, 1989.
3. Rațiu-Suciu, C. ș.a. *Modelarea și simularea proceselor economice*, Editura Didactică și Pedagogică, R.A., București, 1995.
4. Ullman, J.D. *Principles of Database Systems*, Stanford University, 1987.
5. Vandam, A. ș.a. *Fundamentals of Interactive Computer Graphics*, Addison - Wesley, Reading, MA, 1995.
6. *** *Microsoft Visual Basic Version 4.0, Programmers's Guide*, 1995.
7. *** *Microsoft Visual Basic Version 4.0, Professional Features*, 1995.
8. *** *Microsoft Visual Basic Version 4.0, Building Client/Server Applications with Visual Basic*, 1995.
9. *** *Microsoft Visual Basic Version 4.0, LanguageReference*, 1995.
10. *** *Microsoft Visual Basic Version 4.0, Crystal Reports for Visual Basic User's Manual*, 1995.
11. *** *Microsoft Visual Basic Version 4.0, Microsoft Developer Network - MSDN*