

CONTRIBUȚII PRIVIND MODELAREA ȘI CONTROLUL PROCESULUI DE SERVIRE A CLIENȚILOR BANCARI

Radu-George Anghel

e-mail: anghel.radu@gmail.com

Universitatea Politehnica București

Rezumat: În lucrare, se prezintă unele rezultate ale cercetării științifice proprii privind modelarea și simularea sistemelor de servire a clienților din domeniul financiar-bancar. Specificul acestor sisteme de servire din domeniul bancar constă în asocierea fenomenului de risc și finalizarea servirii clienților prin evaluarea cantitativă a riscului de aprobare sau respingere a unei cereri din șirul de așteptare. În încheierea lucrării, sunt prezentate unele rezultate incipiente privind implementarea structurii hardware și testarea experimentală a unui sistem informatic de gestiune a servirii clienților într-o bancă comercială.

Cuvinte cheie: sisteme de servire, șiruri de așteptare, modelare, simulare, testare.

1. Introducere

Sistemul de management al cozilor de așteptare este un nou serviciu informatizat, introdus recent în băncile comerciale. Prin introducerea acestui serviciu, se urmărește fluidizarea circuitului clienților în interiorul unei sucursale și facilitarea accesului rapid al acestora la produsele și serviciile băncii respective. Soluția informatică permite reducerea la minimum a timpului de așteptare a clienților în sucursală, asigură servirea promptă a acestora în funcție de ordinea sosirii, permițând monitorizarea în timp real și reprezintă, totodată, un instrument valoros de marketing pentru bancă. Sistemul de management al cozilor funcționează pe bază de numere de ordine (tip bonuri), pe care clientul și le procură în sucursală din display-uri poziționate la intrare. Apelarea automată și existența unor panouri electronice centralizatoare și locale, situate la fiecare ghișeu, permit optimizarea fluxurilor de operațiuni cu clienții. Ca avantaje ale acestui serviciu, se pot menționa decongestionarea și diminuarea cozilor la ghișee/casierii, măsurarea performanțelor lucrătorilor în relația cu clientul, crearea unui climat de informare adițională a clienților în zonele de așteptare precum și sporirea confidențialității tranzacțiilor, stocarea unor informații privind prelucrarea cererilor de servicii oferite de bancă și categoriile sociale din care banca deține cei mai mulți clienți etc. Instrument de management operațional și de optimizare pe termen mediu și lung, serviciul permite creșterea calității servirii clienților prin diminuarea timpului de așteptare și facilitează analiza gradului de încărcare al lucrătorului bancar și al sucursalelor băncii. Stocarea datelor servirii permite analiza statistică a băncii pe categorii de clienți în vederea promovării celor mai cerute și rentabile servicii.

2. Ipotezele de lucru și parametrii unui algoritm de simulare a sistemului monocanal de servire a clienților unei bănci comerciale

Principalele ipoteze de lucru se referă la tipul distribuției probabilității intervalului de timp dintre două sosiri consecutive de clienți și la durata de execuție a servirii unui client.

Fluxul de cereri la intrare este dat prin funcția de distribuție exponențială (descrie în [1]).

Timpul de execuție τ_j pentru programul de rezolvare a cererii j de către server este o mărime aleatoare, dată prin probabilitatea P_k ($k=1,2,\dots,s$) (există s programe de prelucrare) deoarece sunt s soluționări pentru cererea j . Prin urmare avem:

$$\sum_{i=1}^s P_i = 1$$

În pseudocodul algoritmului de simulare și în descrierea algoritmului, sunt utilizate următoarele notații:

- t_M – Timpul curent de modelare – care crește prin incrementarea cu valori discrete:
 - t_j – momentul lansării cererii în sistem = t , s timpul de execuție al programului apelat de cererea j ;
 - j – timpul curent pentru programul j incrementat discret cu Δ , adică $\bar{\tau}_j = n \Delta$;
 - T – limita superioară a timpului rezervat simulării SS;
 - r – indicele acelei cereri care este prima în șirul de așteptare (prima din coadă);

- τ_j^* - unde $j = (1,s)$ timpul de execuție al programului apelat de cererea j ;
- j - timpul curent pentru programul j incrementat discret cu Δ , adică $\bar{\tau}_j = n \Delta$;
- T - limita superioară a timpului rezervat simulării SS;
- r - indicele acelei cereri care este prima în șirul de așteptare (prima din coadă).

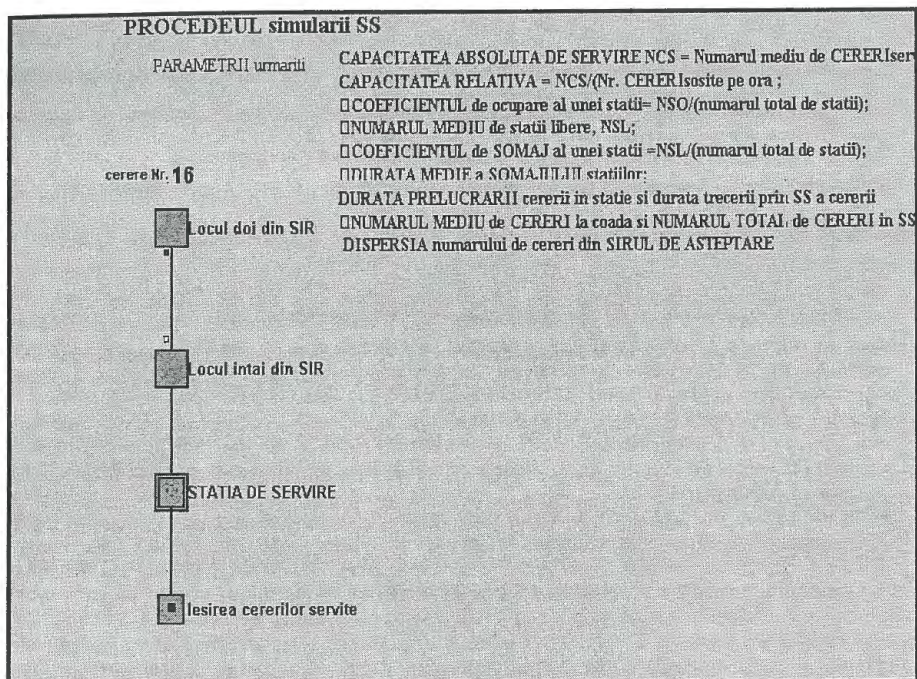


Figura 1. Structura sistemului monocanal simulat

Ca urmare a simulării trebuie obținute estimările:

- t_j^a - intervalul de așteptare = timpul pe care cererea j trebuie să-l aștepte până la începerea servirii sale; dacă momentul inițierii servirii este τ_j^i atunci $\tau_j^a = t_j^i - t_j$;
- $t^{a \text{ rasp}}$ - timp de așteptare a răspunsului la cererea j ;
- $t^{a \text{ rasp}} = t_j^{fs} - t_j$ - (unde t_j^{fs} - este durata finalizării servirii).

Divizarea timpului presupune alocarea unui interval $\Delta =$ constant pentru rezolvarea unei cereri.

Structura sistemului simulat este prezentată în figura 1, iar parametrii sistemului de servire simulat sunt prezentați în figura 2.

- Numărul de console client: 1
- Numărul de console operator: 2
- Numărul mediu clienți pe ora $\Delta \approx .1$
- Gradul de ocupare a unui server cca 68%
- Durata prelucrării (medie): 1,56h
- Numărul mediu de cereri la coadă: 180
- Varianța (dispersia) lungimii cozii :35

Figura 2. Parametrii sistemului simulat

3. Pseudocodul algoritmului de simulare al sistemelor de servire monocanal cu divizarea timpului (SSMCD)

Algoritmul de simulare a SSMCD este compus din două etape: prima etapă se referă la *organizarea clienților în șirul de așteptare*, iar etapa a doua este destinată *simulării servirii adică rezolvării cererilor acestora* [3].

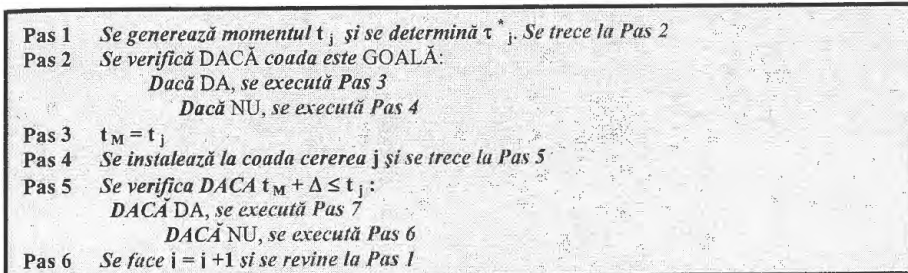


Figura 3. Pseudocodul primei etape a simulării unui SSMCD

Divizarea timpului: se alocă pentru oricare cerere același segment Δ pentru rezolvare. Dacă acest interval s-a scurs și cererea nu a fost rezolvată, ea este pusă din nou la coadă.

La Pas 1, algoritmul de mai sus imită momentele t_j de sosire a cererilor și timpul τ_j^* de execuție al programului apelat de cererea j .

La Pas 2 se verifică dacă în momentul sosirii cererii j coada este goală cu scopul de a stabili care cerere este „prima” la coada (adică stabilește începutul cozii). În ambele situații cererea este instalată în șir la coada.

Pasul 3 a fost introdus pentru adăugarea timpului t_M curent de modelare care crește cu valori discrete până la momentul sosirii t_j .

Pas 4 simulează instalarea.

Pasul 5 verifică dacă în intervalul de divizare Δ nu au sosit cumva și alte cereri (fără Pas 5 se pot pierde o parte din cereri). Astfel, pașii Pas1, Pas 2, ... Pas 6 organizează cererile la coadă.

În figura 4, este prezentat pseudocodul aferent celei de a doua etape din algoritmul de simulare a unui SSMCD. Acesta conține pașii 7 ... 20 ai algoritmului.

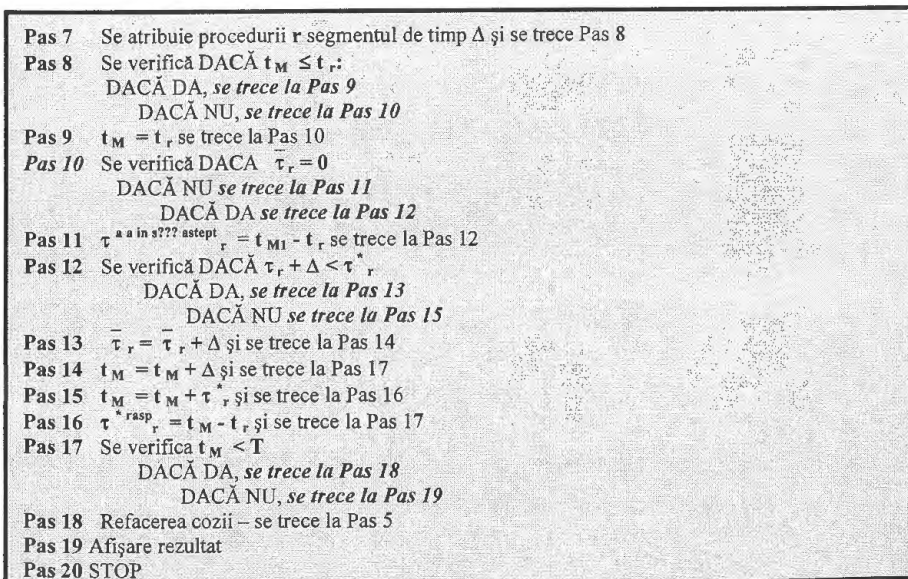


Figura 4. Pseudocodul etapei 2 a simulării unui SSMCD

Pasul 7 alocă procedurii (programului) „r” un segment Δ .

Pasul 8 verifică dacă nu cumva timpul curent de simulare t_M nu a rămas în urma momentului t_r de apariție în sistem a cererii care este „prima” la coadă. Cumva dacă $t_M < t_r$ atunci la **Pas 9** timpul curent al modelării este „tras” până la t_r .

Pasul 10 verifică dacă procedurii cu numărul „r” i s-a alocat măcar un segment de timp Δ .

Dacă **Pas 10** se verifică (adică nu se obține timpul dorit), atunci se adaugă un Δ . Dacă **NU** se verifică **Pas 10**, atunci la **Pasul 11** se calculează durata așteptării τ^a pentru începerea servirii cererii cu numărul „r”.

Pasul 12 verifică dacă programul (procedura) cu numărul „r” apelată de cererea „r” are timp suficient ca în segmentul Δ alocat se poate termina execuția. Dacă condiția de la **Pasul 12** se verifică, atunci programul lansat în execuție de cererea „r” se termină în timp util. Dacă la **Pasul 12** NU se verifică condiția (adică timpul alocat execuției nu este suficient), atunci timpul $\bar{\tau}_r$ de execuție al programului „r” și timpul curent de modelare t_M sunt măriți (**Pașii 13 și 14**), iar dacă **NU** (deci programul reușește să fie executat în intervalul Δ și la **Pasul 15** timpul de modelare?? este mărit până la valoarea realmente necesară pentru terminarea execuției programului „r” (acest interval poate fi mai mic ca Δ deoarece $(\tau^a - \bar{\tau}_r) \leq \Delta$). Prin aceasta, se realizează o economisire a timpului de modelare.

Pasul 16 calculează intervalul de timp τ^{resp}_r – după care cererea rezolvată (r) obține răspunsul (rezolvarea).

Pasul 17 verifică atingerea cotei superioare T a intervalului de modelare. Dacă această cotă nu este atinsă, atunci se reconstruiește coada prin deplasarea cu un loc a cererilor. Aceasta se obține prin utilizarea registrului de deplasare din figura 5, unde este afișat și programul de generare a unui șir de semnale pseudoaleatoare binare [2].

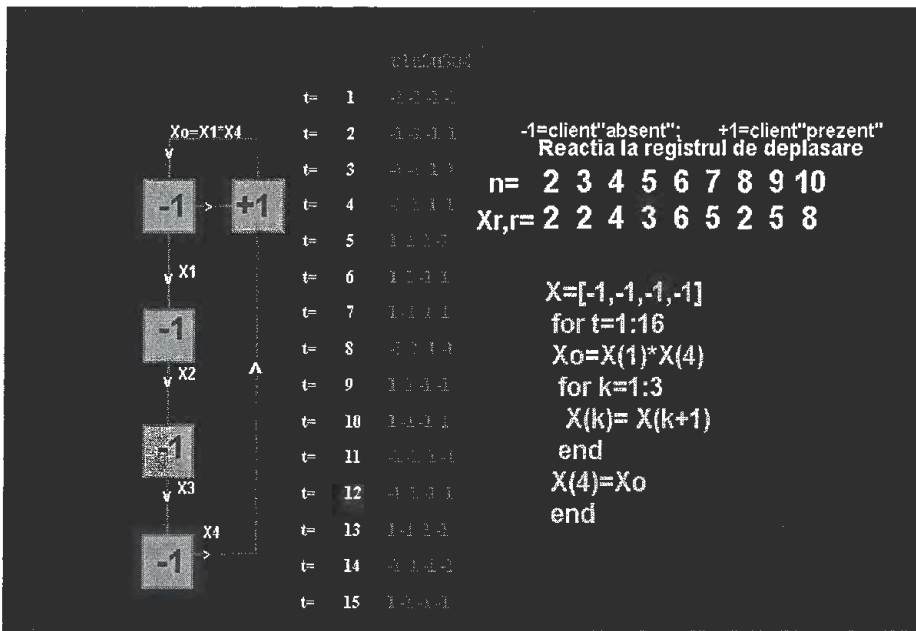


Figura 5. Generarea evenimentelor sosirii clienților folosind registrul de deplasare

Această cercetare științifică a urmărit verificarea experimentală a unor aspecte pur teoretice, prezentate anterior, privitoare la sistemele de servire a clienților în cadrul unei Bănci Comerciale la care am avut acces în perioada doctoraturii. În continuare, sunt prezentate câteva aspecte privind arhitectura sistemului implementat, precum și unele rezultate experimentale ale testării „pe viu” a sistemului implementat în cadrul activității de cercetare, în perioada doctoraturii.

4. Structura și arhitectura sistemului informatic interactiv de gestiune și informare a clienților(SAGIC) dintr-o bancă comercială

Acesta este un subsistem al unui sistem informatic bancar, care nu funcționează independent, ci a fost integrat, din punct de vedere hard-soft, în structura existentă a sistemului informatic principal al băncii. **SAGIC** este un sistem cu mai multe console pentru clienți în vederea comunicării interactive cu sistemul, și mai multe stații de servire a cererilor în ordinea sosirii la coadă și reintroducere în coadă a cererilor care nu au rezistat la testul de risc. Spre exemplu, aceste cereri sunt reîntoarse clienților cu recomandări concrete privind revizuirea cererii și ameliorarea condițiilor care afectează riscul acordării creditului. În figura 6, este prezentată, ca exemplu, structura unui astfel de sistem prevăzut cu trei console-client și două servere.

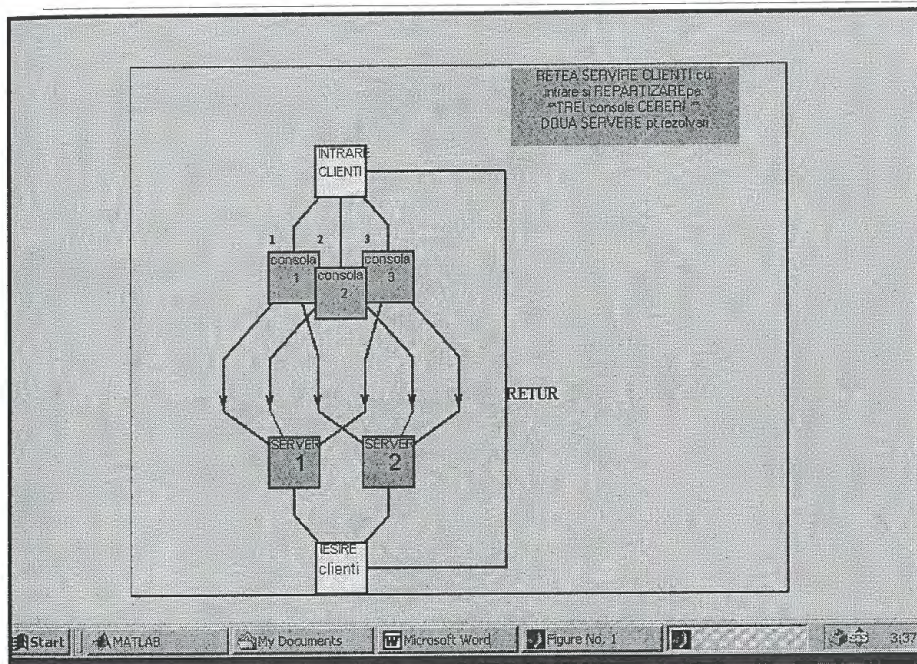


Figura 6. Structura unui SAGIC cu 3 console-client și 2 servere

Acest sistem este prevăzut cu un circuit de RETUR a cererilor cu probleme.

- Numărul maxim de console –client gestionate de UC: 10 console client
- Numărul maxim de console-operator de tip WS: 22 stații
- Prelucrări de informații și afișaj rezultat la cerere:
 1. NR de clienți pe fiecare oră (min, mediu, max)
 2. TOTAL clienți pe zi
 3. Bargraf clienți pe intervale de timp
 4. Afișare timp de așteptare (min, mediu, max)
 5. Afișare timp de prelucrare (min, mediu, max)

Figura 7. Date, parametri și facilități ale sistemului implementat

Arhitectura sistemului testat este prezentată în figura 8, unde terminalele aflate la dispoziția clienților sunt plasate în compartimentele B#1.....,B#n unde cu $i = 1,2,\dots,n$, respectiv a, \dots, z s-au notat panourile electronice de afișare (1,2,...,n& a,b,...,z) de cereri la coadă.

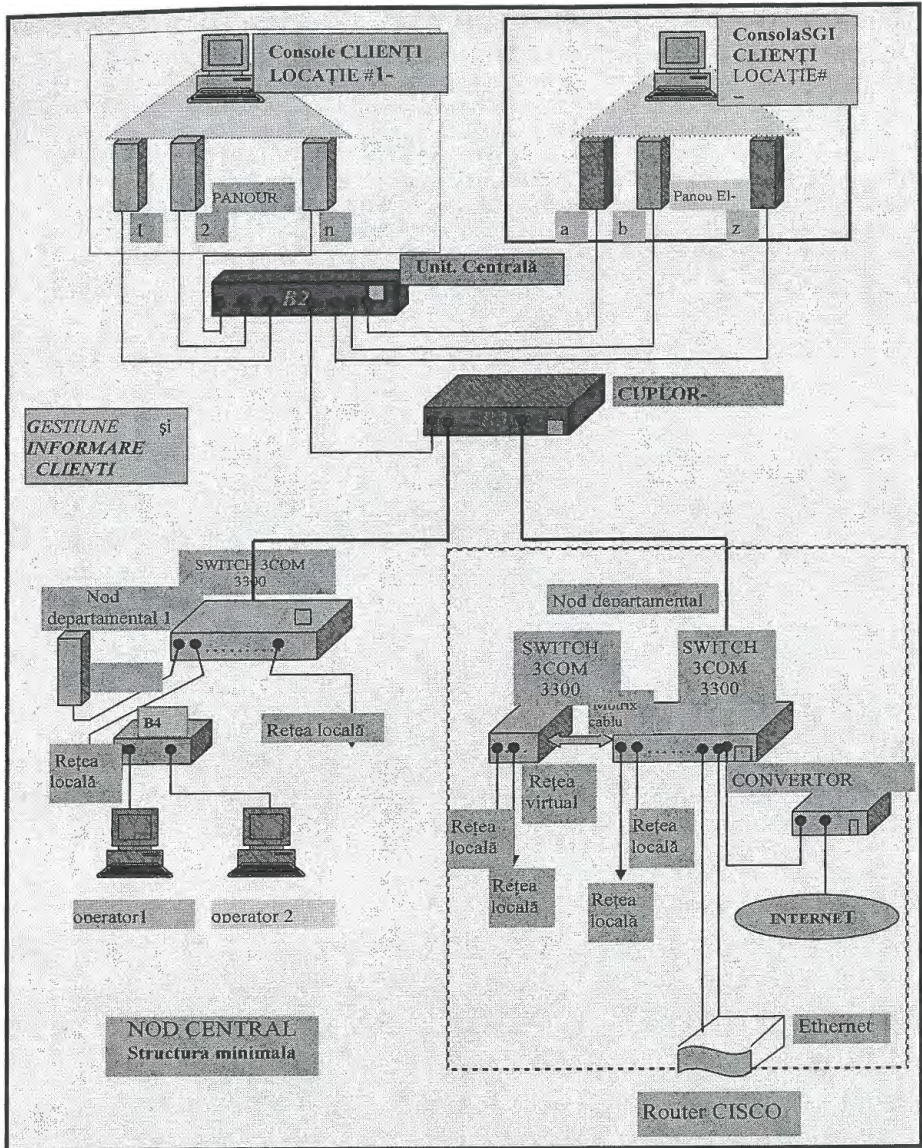


Figura 8. Arhitectura SAGIC(B1,B2 ;B3;B4) și integrarea în sistemul informatic central

În figura 8, sunt enumerate principalele blocuri B1, B2, B3, B4 componente ale acestui sistem. Funcționalitatea acestora este următoarea:

- B#1.....#n: compartimentele consolă-client care conțin monitoarele și tablourile electronice 1, 2, ...n-a, b, ... ,z, pentru afișarea informației-operator transmisă clientului în cadrul comunicației interactive om-calculator;
- B2 Unitatea Centrală este practic un calculator, care facilitează afișarea pe tablourile electronice, stocarea și prelucrarea informației-client;
- B3 cuplor-rețea - asigură legătura duplex la rețeaua locală și INTERNET;

- B4 cuplur Stații WS - operator1 și stațiile operator1, operator2 asigură comunicația duplex între operatorii departamentali din rețeaua locală de calculatoare și consolele-client.

Rezultatele pentru cererile preluate în stații-operator, (denumite *operator-1* și *operator-2*), de tratare a cererilor se face pe canalul direct. Pe canalul invers, rezultatul tratării cererii este afișat pe monitorul consolei-client, dar și pe panoul electronic, aferent departamentului căruia îi este adresată cererea. Pentru oricare cerere, este alocat același interval reglabil de timp Δ (nu este vorba de timpul necesar tratării cererii !). Dacă cererea nu s-a rezolvat de către *operator - i* în intervalul Δ de timp, cererea respectivă este reinstalată prin canalul de invers la sfârșitul cozilor ce urmează a fi procesate.

Sistemul descris a fost testat în cadrul unei bănci comerciale. Informațiile culese prin stațiile operator 1 și operator 2 sunt disponibile inițial în memoria RAM a unității centrale, ele sunt disponibile pe și pot fi citite la cerere printr-un program ce funcționează sub Internet Explorer. La indicarea datei (spre exemplu 2006.07.07) sunt selectate cele două pagini (bargraful din figura 9 și tabelul din figura 10).

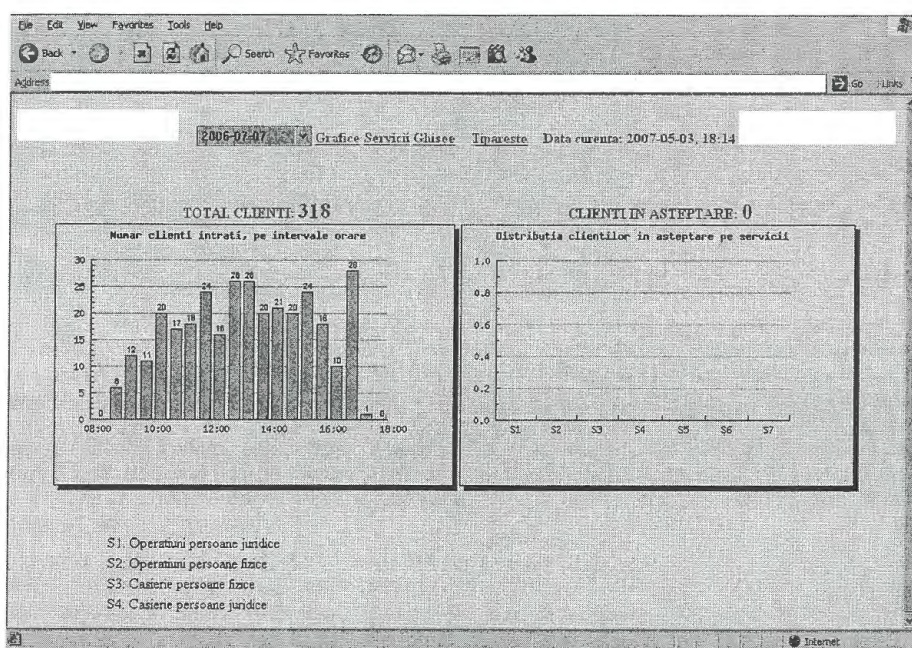


Figura 9. Construirea și afișarea bargrafurilor la cerere pentru ziua de 07-07-2006 (data citirii 05-03-2007)

La citire, este adăugată data și ora la care s-a produs cererea pentru obținerea acestor informații (de exemplu, în figura 7.5 este indicată data 2007/05/03 ora 18 și 14 min., iar în următoarea figura 7.6 este indicată data 2007/05/03 ora 18 și 16 min.

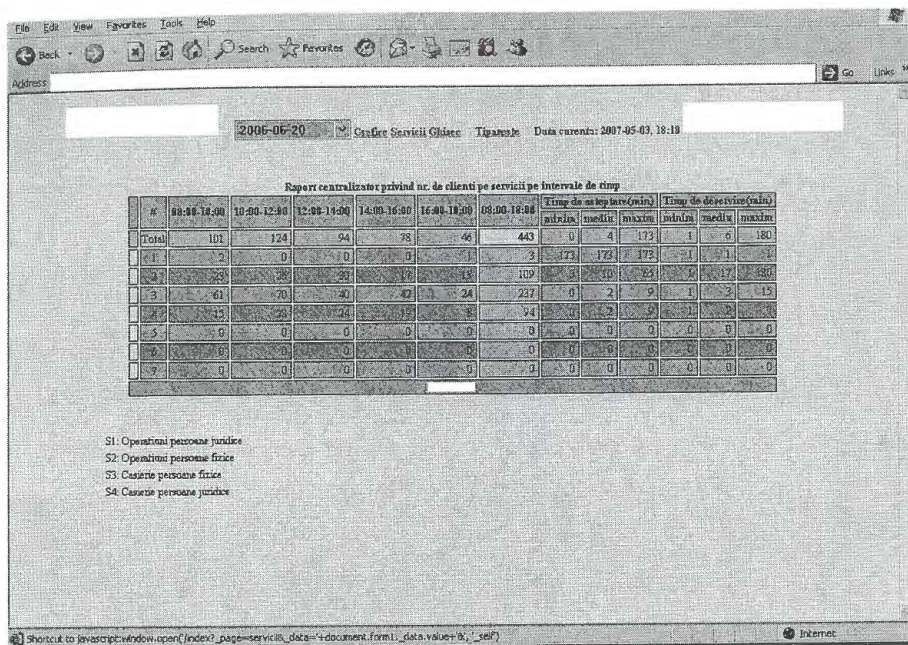


Figura 10. Stocarea și afișarea la cerere a datelor pentru ziua de 20-06-2006

În perioada testării experimentale, au fost urmăriți și stabiliți parametrii sistemului și condițiile de funcționare, rezultând valorile prezentate în figura 10.

6. Concluzii

Principalele contribuții ale lucrării se referă la modelarea și simularea unui sistem destinat managementului cozilor de așteptare ale clienților unei bănci comerciale. Prin introducerea acestui serviciu, se urmărește fluidizarea circuitului clienților în interiorul unei sucursale și facilitarea accesului rapid al acestora la produsele și serviciile băncii respective. Soluția arhitecturală, propusă în lucrare, prezintă particularitatea integrării acestui subsistem în nucleul informatic central al băncii comerciale respective.

Bibliografie

1. ANGHEL, R.: Evaluarea parametrilor unor rețele complexe din domeniul financiar bancar. Revista Română de Informatică și Automatică, vol. 15, nr. 4, 2005, pp. 43-56.
2. TERTIȘCO, M., P. STOICA: Identificarea sistemelor, Ed. Academiei Române, 1980.
3. STĂNCULESCU, FL.: Modelling of High Complexity Systems With Applications, WIT-press, 2005.