

DEZVOLTAREA APLICAȚIILOR CU BAZE DE DATE ȘI GENERAREA FIȘELOR TEHNOLOGICE

M. Urdea

Universitatea Transilvania din Brașov

INTRODUCERE

Lucrarea de față face o prezentare generală a bazelor de date tehnologice, ocupându-se în final de datele utile ce permit desenarea fișelor tehnologice, cu exemplificare, a fișelor tehnologice pentru inelul interior al rulmentului radial axial cu role conice.

1. APLICAȚIILE CU BAZE DE DATE

Aplicațiile cu baze de date pot fi privite din două puncte de vedere: cel al utilizatorului și cel al proiectantului [1].

Din punct de vedere al utilizatorului aplicația cere date prin elemente de interfață, urmând ca pe baza acestora să obțină rezultate pe ecran disc sau imprimantă.

Din punct de vedere al proiectantului aplicația este văzută mult mai complex. Datele cu care operează aplicația sunt depozitate în una sau mai multe baze de date în funcție de aceasta definindu-se și programele aplicației. Aceste programe sunt diferențiate astfel:

- Programe de introducere de date, prin intermediul cărora sunt actualizate bazele de date. Aceste date se introduc fie de către utilizator, în ecrane de introducere fie se importă de la alte surse exterioare (fișiere);

- Programe de prelucrare a datelor din bazele de date, necesare obținerii rezultatelor;

- Programe de extragere de date din bazele de date, sub forma de rapoarte statistice, informații curente din baza de date, sub diferite forme de prezentare exterioară.

Tehnica de procesare a datelor a evoluat de-a lungul anilor, principalul obiectiv fiind accesul cât mai rapid, în mod aleator, la baza de date. Procesarea "Batch", utilizată inițial, a fost abandonată în favoarea tehnicii OLTP – *On Line Transaction Processing*, deoarece aceasta din urmă oferă interactivitate. Aspectul interactiv al OLTP a avut un impact semnificativ asupra modului de utilizare și organizare a bazelor de date.

Tehnica *Data mining* aduce inovații în ceea ce privește modul de analiză a datelor și de interpretare a rezultatelor. Această tehnică este focalizată pe

vizualizarea datelor – de unde rezultă și nevoia de computere mai rapide și facilități grafice deosebite.

Tehnica analitică de căutare a informațiilor, cunoscută sub denumirea de *Pattern matching* înseamnă capacitatea de a detecta și selecta datele.

Bazele de date au un rol hotărâtor în dezvoltarea sistemelor informatice. Există în momentul actual o tendință către stocarea centralizată a datelor într-o așa numită *Engineering Database*, respectiv o bază de date comună. Lucrări actuale de cercetare au ca scop realizarea unor platforme de integrare pentru diverse sisteme.

O bază de date este un ansamblu de informații structurate în înregistrări *records* compuse din câmpuri *fields*. Înregistrările pot fi înțelese drept linii ale unui tabel, iar câmpurile drept coloane. Înregistrările au lungime fixă alocându-se un spațiu și atunci când unele câmpuri nu sunt completate cu date (ele sunt completate cu blancuri sau zerouri, în mod implicit). Fiecare câmp poate fi de tip numeric, caracter, logic, sau dată calendaristică. În plus se mai poate adăuga și așa numitele câmpuri "memo" care conțin text bogat asociat unei poziții de câmp. Ordinea înregistrărilor este mai puțin importantă, deoarece sistemul care gestionează baza de date (SGBD-ul) excelează în posibilități de sortare și ordonare a înregistrărilor și informațiilor, în general, conținute în baza de date.

SGBD permite o întreagă paletă de operații dintre care: vizualizarea ordonată și/sau filtrată a înregistrărilor după orice câmp sau combinație de câmpuri (interogare), asocierea a două sau mai multe baze de date prin câte un câmp cu date comune (baza de date relaționale), extragerea de rapoarte tip listă sau tip etichetă privind anumite câmpuri dintr-o serie filtrată și sortată de înregistrări, crearea și utilizarea mijloacelor de introducere/actualizare comodă și sigură a datelor, asocierea unor programe utilizator prin care să se specifice prelucrări particulare ale bazei de date (programare).

Toate operațiile trebuie făcute în condiții de maximă securitate a datelor, atât pe calculatoare izolate, cât și pe stații ale unei rețele de calculatoare. Simpla existență a unei structuri de bază de date nu este așa de valoroasă dacă nu există un program de valorificare a ei și dacă nu este mereu actualizată și întreținută.

Un server de date este un program care se execută pe un calculator server și permite inițializarea accesarea și întreținerea unei baze de date. Într-o arhitectură client/server, serverele de baze de date sunt aproape întotdeauna servere SQL.

2. BAZE DE DATE ÎN PROIECTAREA TEHNOLOGICĂ

Bazele de date suport asigură legătura între programele de proiectare și sistemele de gestiune a bazelor de date folosind module specializate. O aplicație a acestor baze de date este utilizată de un mediu de proiectare asistată și anume *AutoCAD* [1].

Bazele de date cu elemente standard conțin organe de mașini pentru asamblare, rulmenți, elemente pentru matrițe etc., utilizând cataloagele unor firme consacrate. Bazele de date sursă conțin informații despre componente (dimensiuni, material, preț, furnizor), folosite la proiectarea parametrizată. Utilitatea acestor baze de date a fost evidențiată prin produsele *DesignCAD* și *Microstation*.

Bazele de date proprii sunt folosite la mărirea productivității proiectării prin realizarea automată a unor operații ca: modelare 3D, cotare, simbolizare, conectare la alte programe CAD de inginerie și fabricație asistată. Exemple în acest sens sunt bazele de date folosite în: *FastCAD*, *SolidEdge*, *Euclid*.

Exemple de baze de date specializate în domeniul proiectării îl constituie și băncile de calcul (exemple *BCH* și *BCP*). În componența acestora intră formulele, algoritmi și metode necesare rezolvării unei anumite teme de proiectare [1].

Producătorii și utilizatorii din categoria firmelor cu mare pondere pe piața mondială și-au creat propriile baze de date. Bazele de date permit o bună organizare a stocurilor de materiale, scule tipizate etc.

În majoritatea cazurilor, baza de date va fi citită de programul asociat și va fi prelucrată conform acestui program, rezultând în final o altă bază rezultantă ce va putea fi prelucrată într-un program de grafică.

Datele inițiale, implicate în orice proiectare tehnologică asistată, sunt divizate în două categorii: date inițiale fixe (de nivel 1) și date inițiale variabile (de nivel 2). Datele inițiale fixe sunt acele date care sunt utilizate în rezolvarea tuturor problemelor în plaja de valabilitate a sistemului program. Datele inițiale variabile sunt acele date care particularizează fiecare caz în parte.

Datele inițiale fixe vor alcătui baza de date a sistemului fiind organizate în fișiere iar cele

variabile vor fi introduse în calculator în majoritatea cazurilor de la tastatură utilizându-se tehnici tradiționale meniuri sau casete de dialog. Datele fixe se pot "depozita" în fișiere tip text editate în editoare de texte clasice, pot fi organizate în tabele în *Excel*, *Access*, *dBASE*, *SQL* sau pot fi organizate în programe specializate.

Datele variabile se introduc la rularea unui program accesibil și prietenos care în majoritatea cazurilor este realizat utilizând un limbaj evoluat (*C++*, *Pascal*, *LabView*, *AutoLISP*, *Matlab* etc). Acest program creat special pentru procesarea datelor, prelucrează o parte din datele fixe și face calcule, are blocuri de decizie, oferă informații și listează date pe ecran sau la imprimantă.

3. ELEMENTE GENERALE DE PROGRAMARE TEHNOLOGICĂ

În aplicațiile tehnologice datele trebuie reprezentate și organizate în conformitate cu disponibilitățile și legitățile limbajului de programare sau sistemului software utilizat. Datele se utilizează în programe sub formă de constante și variabile. Constanta are asociată o valoare, numită *valoarea constantei* ce este mereu aceeași pe tot parcursul execuției. Variabila reprezintă o locație de memorie care poate găzdui un element dintr-o mulțime dată, numită domeniu de definiție al variabilei. Datele pot fi clasificate în *date elementare* și *date structurate* (compuse).

Datele elementare sunt obiecte indivizibile ale limbajelor de programare a căror structură internă nu este accesibilă utilizatorului.

Datele structurate sunt alcătuite din mai multe componente, toate accesibile utilizatorului. Componentele pot fi date elementare sau la rândul lor structurate, între ele existând anumite relații, specifice structurii respective.

În general limbajul C poate fi utilizat la prelucrarea datelor. Cu ajutorul acestui limbaj se poate prelucra o bază de date, rezultând programe de calcul compilate și anume fișiere executabile ce pot fi ușor accesate de utilizator.

Limbajele favorizează modularizarea programelor, prin utilizarea proiectelor. De exemplu un proiect C este un program realizat ca ansamblu de module (pachete de programe) în format sursă sau precompilate.

Posibilitățile de lucru cu adrese de date și de funcții (pointeri), efectuarea de operații aritmetice și comparații, posibilitatea de a structura cuvintele de memorie (câmpuri de biți) și de a efectua operații la nivel de bit, oferă limbajului C calități deosebite necesare și în programarea inginerescă. C câștigă

în plus construcțiile necesare pentru aplicarea principiilor programării orientate spre obiecte (POO). Programele prelucrează datele inițiale fixe și datele variabile, folosesc o interfață ușor de înțeles și manevrat creată special pentru utilizator, răspunzând prin blocurile de decizie la restricțiile impuse de proiectant.

Spre deosebire de C, care impune gruparea declarațiilor generale la începutul programului principal, sau la începutul unui bloc, limbajul C++ permite declarații de date inițiale în fișiere *header* (*antet*). Programele mari pot fi realizate din mai multe programe sursă, compilate separat, legând apoi împreună diferitele module obiect. Pentru a realiza mai multe module este necesară crearea unui *fișier proiect* în care se specifică fișierele componente ale programului. La lansarea proiectului are loc compilarea pe rând a fiecărui fișier sursă creându-se fișierul obiect corespunzător, în final rezultă un fișier executabil cu extensia "exe" care lansează în execuție programul obținut.

Majoritatea programelor de acest gen prelucrează date, realizează calcule matematice, prelucrează blocuri de decizie conform unui algoritm stabilit din timp. Bazele de date se creează într-un editor simplu de texte și se apelează cu ajutorul unor funcții de citire. Programele pot utiliza funcții grafice, de vizualizare, în cazul rulării sub MS-DOS sau pot rula sub Windows, folosind facilitățile Windows.

În finalul lansării programelor funcție de specificul și interesul programatorului pot rezulta fișiere de date care pot fi de exemplu tip *Script* sau *DXF* putând fi recunoscute de un program grafic CAD (Fig. 1).

Încărcarea fișierelor *Script* și *DXT* în AutoCAD se va face cu comanda *Script* și respectiv *Dxfout*. Fișierele *Script* sunt fișiere text care conțin numele unor comenzi AutoCAD, precum și răspunsurile la dialogul comenzilor. Formatul *DXF* (fișier de schimb de desene) a devenit standardul acceptat pentru traducerea fișierelor între diferite programe CAD. Fișierele create cu formatul *DXF* sunt scrise în codul *ASCII* standard, fișierul putând fi citit de orice calculator. Această metodă (*Script / DXF*) se folosește atunci când nu este necesară reprezentarea la scară. Din desenele suport se vor crea fișiere *DXF* folosind comanda *Dxfout*.

Fișierele *DXF* se vor crea în directorul în care există și trasajele inițiale. Fișierele *DXF* se vor vizualiza cu aplicația *Wordpad* și vor fi salvate cu aceeași denumire dar cu extensia *X00*. Din aceste fișiere se vor șterge ultimele patru rânduri, ele fiind astfel pregătite să primească în continuare date tip *DXF*, ce vor putea completa în final desenele

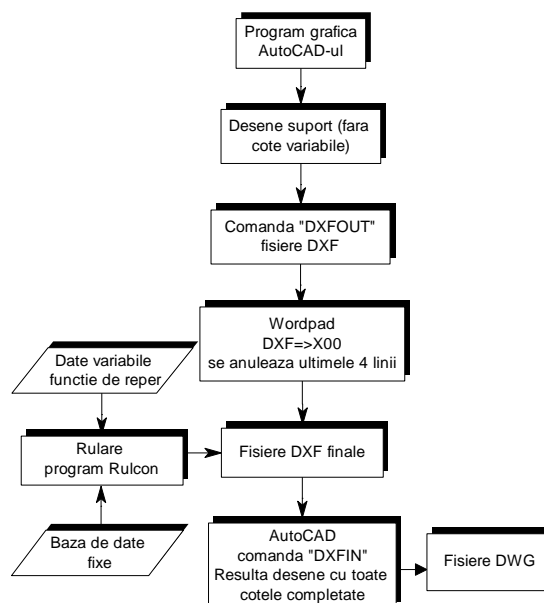


Figura 1. Prelucrarea datelor cu obținerea unor desene tehnologice.

considerate. Fișierele *DXF* vor fi recunoscute de un program grafic CAD, cum este AutoCAD-ul [2,3].

4. GENERAREA FIȘELOR TEHNOLOGICE

4.1 Creație software pentru desenarea rapida a fișelor tehnologice

Pentru exemplificare se prezintă pachetul de programe **Rulcon** organizat în proiectul **Rulcon**. Acesta permite introducerea datelor de intrare și prelucrarea unei baze de date existente, precum și obținerea unor fișiere *DXF* care generează în AutoCAD fișele tehnologice corespunzătoare rulmentului radial axial cu role conice (Fig. 2) dintr-o serie specială L [4].



Figura 2. Rulment radial axial cu role conice.

Programele sursă sunt în număr de patru și anume: **conmain.cpp**, **condxf.cpp**, **dxl.cpp**, **dtos.cpp**.

Conmain cuprinde funcțiile de citire a fișierelor de date pentru cele doua inele, majoritatea casetelor de dialog (alegerea filei din planul de operații), funcțiile de calcul, posibilitatea de introducere a datelor în cazul în care datele nu sunt cuprinse în fișierele de date, *inel_ex.txt* și

inel_int.txt. În concluzie, programul Conmain.cpp are rol de organizator de date și de decizie.

Programul **condxf.cpp** permite listarea datelor pe ecran sau la imprimanta, citește fișierele DXF trunchiate și conține funcțiile de generare a fișierelor DXF noi cu valori completate funcție de reper.

Dxf.cpp este un fișier deosebit de important în prelucrarea fișierelor DXF. Acest program completează pas cu pas lista lungă a fișierelor DXF, funcționând conform codificării DXF și fiind compus din funcții de citire și scriere. În plus Dxf.cpp gestionează înălțimea și unghiul de scriere, factorul de scară, stilul de scriere, etc. Programul "Dxf.cpp" poate utilizat și la alte aplicații nefiind specializat pentru o anumită problemă.

Dtos.cpp permite scrierea cifrelor zecimale cu virgulă, făcând transformarea din "." în ",".

Fișierele Dxf.cpp și Dtos.cpp sunt dependente de fișierele antet **Dxf.h** și **Dtos.h**, hotărâtoare în completarea fișierelor X00 cu datele prelucrate.

La lansarea pachetului de programe sunt căutate fișierele cu extensia X00 ce trebuie completate cu valorile calculate și prelucrate în forma DXF.

Toate fișierele text și fișierele X00 vor exista în catalogul curent, de unde se rulează și fișierul executabil **Rulcon.exe** rezultat la compilare.

4.2 Desenarea fișierelor tehnologice

S-a ales, pentru exemplificare, inelul exterior, component al rulmentului radial axial cu role conice. Din fișa tehnologică, figura 3 prezintă fila șablon de control final, fără cotele variabile ce depind de reper.

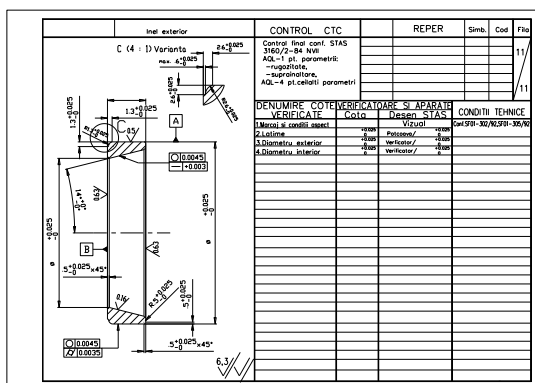


Figura 3. Control final șablon pentru inel exterior

La rularea aplicației rezulta un fișier DXF ce depinde de opțiunea utilizatorului. Acest fișier deschis în AutoCAD, este o filă din fișa tehnologică corespunzătoare inelului interior sau exterior al rulmentului radial axial cu role conice.

Spre exemplificare, se prezintă fila cu control final corespunzătoare inelului exterior.

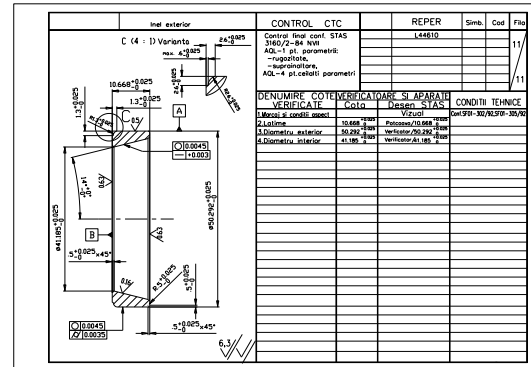


Figura 4. Control final, inel exterior

4.3 Aplicația Rulcon

La rularea aplicației Rulcon cere o parolă pentru deschiderea programului, oferă informații utilizatorului, cere datele de intrare, apelează baza de date, existând posibilitatea de introducere de date în cazul unor repere noi (Fig. 5a), permite utilizatorului alegerea componentului pentru prelucrarea fișierelor DXF, listează datele și creează fișierele DXF (Fig. 5b).

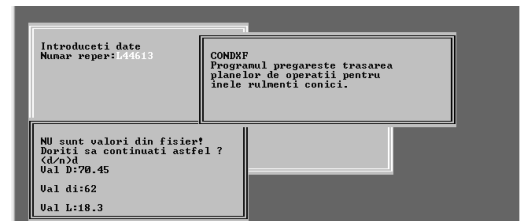


Figura 5a. Rulcon.exe, pentru introducerea de date noi

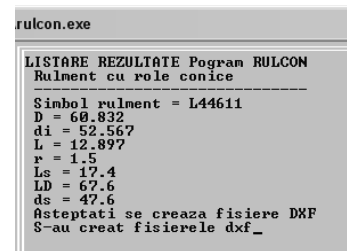


Figura 5b. Rulcon.exe, listarea datelor și realizarea fișierelor DXF

Figura 6 prezintă organizarea proiectului Rulcon cu fișierele componente, prezentate în capitolul 4.1.

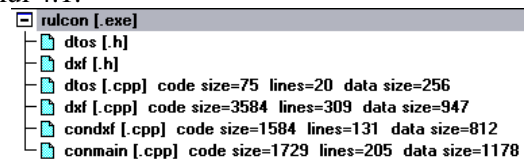


Figura 6. Proiectul Rulcon

Sunt prezentate în continuare funcțiile importante din Proiectul Rulcon, pentru prelucrarea datelor, pentru inelul exterior.

Din programul **Conmain.cpp**:

```
main()
{ deschidere();
Help();
informatii();//...etc
FILE *fp;
int p,m1;
fp=fopen("inel_ex.txt","r");
//"r" deschide un fisier pentru citire
if(fp==NULL)
{ cputs("\r\n Nu se poate deschide fisierul");
return(0); }
window(3,5,37,20);
textbackground(LIGHTGRAY);
clrscr();
border();
window(4,6,36,19);
textcolor(BLUE);
gotoxy(1,1);
cputs("\r\n Introduceti date");
gotoxy(1,2);
cputs("\r\n Numar reper:"); gets(V1);
gotoxy(1,3);
do {
p=fscanf(fp,"%s%f%f%f",V1a,&D,&di,&L);
m1 = memcmp(V1,V1a,6); //comparare
if(p==EOF)
{ window(3,15,37,25);
textbackground(LIGHTGRAY);
clrscr();
border();
window(4,16,36,24);
textcolor(BLUE);
cputs("\r\n NU sunt valori din fisier!");
cputs("\r\n Doriti sa continuati astfel ?");
cputs("\r\n (d/n)");
if(getche()=='d')
{ cputs("\r\n Val D:"); scanf("%f",&D);
cputs("\r\n Val di:"); scanf("%f",&di);
cputs("\r\n Val L:"); scanf("%f",&L);
calcule();
rezultate();
toDXFtul();
if(sfarsit()==0) return 0; }else return 0; }
} while(m1 !=0);
calcule();
rezultate();
toDXFtul();
fclose(fp);
return 0; }
```

Din programul **Condx.cpp**:

```
void toDXFtul(void)
```

```
{ cputs("\r\n Asteptati se creaza fisiere DXF");
getche();
char buf[256];
DXF dxf;
errorCode = 0;
fnsplit(umeDesen,buf+0,buf+3,buf+69,NUL
L);//desface in comp. umeDesen
*( buf+69+7 )='\0'; //trunchiaza la 7
caractere
strcat( buf+69, "X" ); //adauga X la denumire
fnmerge(
umeDesen,buf+0,buf+3,buf+69, ".DXF");
//constr. o noua cale
strcpy(buf, caleCuz);
if(*buf) strcat(buf,"\\");
strcat(buf,"rul_ext11.X00");
*(strchr( umeDesen, '.')-1)='K';
if(dxf.open(umeDesen, buf)) {errorCode =
5; return; }
sprintf(buf,"%s",V1);
dxf.textC(buf,"iso",190.0,40.5, 0.0);
sprintf(buf,"%f",D);
dxf.text(buf,"iso",150.09,192.50, 90.0);
sprintf(buf,"%f",di);
dxf.text(buf,"iso",34.08,187.47, 90.0);
sprintf(buf,"%f",L);
cputs("\r\n S-au creat fisierele dxf");//etc...
getche();
if(dxf.close()) {errorCode = 5; return; } }
```

Bibliografie

1. **Zamfir, N.** Bază de date destinată proiectării tehnologice asistate de calculator. Rezumatul tezei de doctorat, Universitatea "Transilvania" din Braşov, 2000.
2. **Urdea, M.** Crearea unor bănci de date pentru proiectarea constructivă și tehnologică a pieselor tip bucușă, teza de doctorat (2003).
3. **Urdea, M.** Dezvoltarea unei aplicații în C++ pentru prelucrarea bazei de date a rulmenților radial-axiali cu două rânduri de bile, publicație în revista *Recent*, Universitatea Transilvania din Braşov, Vol.6 (2005) ISSN 1582-0246.
4. **Urdea, M.** A computerized method for rapid drawing for the tapered roller bearing in several dimensions, XI th International Conference in the knowledge based organization Conference proceedings IX, 23-25 Nov. 2006, Academia forțelor terestre Nicolae Balcescu in colaborare cu Universitatea Lucian Blaga Sibiu, p. 216-221, ISBN: 973-7809-51-3, 978-973-7809-51-3.