

I. CONCEPTE ALE BAZELOR DE DATE RELAȚIONALE

1.1 Definiții

1.2 Niveluri de abstractizare a datelor

1.3 Componente ale bazelor de date relaționale

1.4 Proiectarea bazelor de date relaționale. Etape. Normalizarea bazelor de date

1.1 Definiții

Baze de date

O bază de date este o colecție de informații interrelaționate gestionate ca o singură unitate. Această definiție este intenționat foarte largă, deoarece există mari diferențe între concepțiile diferiților producători care pun la dispoziție sisteme de baze de date. De exemplu, Oracle Corporation definește o bază de date ca fiind o colecție de fișiere fizice gestionate de o singură instanță (copie) a produsului software pentru baze de date, în timp ce Microsoft definește o bază de date SQL Server ca fiind o colecție de date și alte obiecte. Un *obiect* al bazei de date este o structură de date denumită stocată în baza de date, cum ar fi un tabel, o vizualizare sau un index.

Sisteme de gestiune a bazelor de date

Un *sistem de gestionare a bazei de date (DBMS - database management system)* este un produs software furnizat de producătorul bazei de date. Produse software precum Microsoft Access, Microsoft SQL Server, Oracle Database, Sybase, DB2, INGRES, MySQL și PostgreSQL fac parte din categoria DBMS sau, mai corect, *DBMS relaționale (RDBMS)*.

Sistemul DBMS pune la dispoziție toate serviciile de bază necesare pentru organizarea și întreținerea bazei de date, inclusiv următoarele:

- Transferarea datelor în și din fișierele fizice de date, în funcție de cerințe.
- Gestionarea accesului concurențial la date al mai multor utilizatori, inclusiv prevenirea conflictelor care ar putea fi cauzate de actualizările simultane.
- Gestionarea tranzacțiilor, astfel încât toate modificările făcute asupra bazei de date printr-o tranzacție să fie executate ca o singură unitate. Cu alte cuvinte, dacă tranzacția reușește, toate modificările efectuate de tranzacție sunt înregistrate în baza de date; dacă tranzacția eșuează, nici una dintre modificări nu este înregistrată în baza de date. Totuși, unele sisteme RDBMS nu asigură suportul pentru tranzacții.
- Acceptă un *limbaj de interogare*, care reprezintă sistemul de comenzi folosit de utilizator pentru a obține date din baza de date. SQL este principalul limbaj folosit pentru sistemele DBMS relaționale.
- Funcții pentru salvarea bazei de date și pentru refacerea bazei de date în urma erorilor.
- Mecanisme de securitate pentru împiedicarea accesului neautorizat la date și modificarea acestora.

Baze de date relaționale

O *bază de date relațională* este o bază de date care respectă modelul relațional, dezvoltat de Dr. E.

F. Codd. Modelul relațional prezintă datele sub forma familiarelor tabele bidimensionale, similar cu o foaie de calcul tabelar Excel. Spre deosebire de o foaie de calcul tabelar, nu este obligatoriu ca datele să fie stocate într-o formă tabelară, iar modelul permite și combinarea tabelelor (*crearea uniunilor (joining)*, în terminologia relațională) pentru formarea vizualizărilor, care sunt prezentate tot ca tabele bidimensionale. Flexibilitatea extraordinară a bazelor de date relaționale este dată de posibilitatea de a folosi tabelele independent sau în combinații, fără nici o ierarhie sau secvență predefinită în care trebuie să se facă accesul la date.

1.2 Niveluri de abstractizare a datelor

Într-un sistem informatic ce utilizează baze de date, organizarea datelor poate fi analizată din mai multe puncte de vedere și pe diferite niveluri. De obicei, abordarea se face pe trei niveluri: intern, conceptual și extern (fig. 1.1).

- Nivelul fizic (intern)

Structura datelor este descrisă foarte detaliat, fiind accesibilă numai specialiștilor (ingineri de sistem, programatori în limbaje de asamblare sau alte limbaje apropiate de „mașină”). Cele două părți principale ale bazei la acest nivel sunt:

1. un set de programe care interacționează cu sistemul de operare pentru îmbunătățirea managementului bazei de date;
2. fișierele stocate în memoria externă a calculatorului.

Fișierele ce conțin datele propriu-zise sunt alcătuite din articole sau înregistrări cu format comun. La acest nivel, structura bazei de date se concretizează în *schema internă*.

- Nivelul conceptual (global)

Este nivelul imediat superior celui fizic, datele fiind privite prin prisma semanticii lor; interesează conținutul lor efectiv, ca și relațiile care le leagă de alte date. Reprezintă primul nivel de abstractizare a lumii reale observate. Obiectivul acestui nivel îl constituie modelarea realității considerate, asigurându-se independența bazei față de orice restricție tehnologică sau echipament anume. Întreaga bază de date este descrisă prin intermediul unui număr restrâns de structuri. Toți utilizatorii își exprimă nevoile de date la nivel conceptual, prezentându-le administratorului bazei de date, acesta fiind cel care are o viziune globală necesară satisfacerii tuturor cerințelor informaționale. La acest nivel, structura bazei de date se concretizează în *schema conceptuală*.

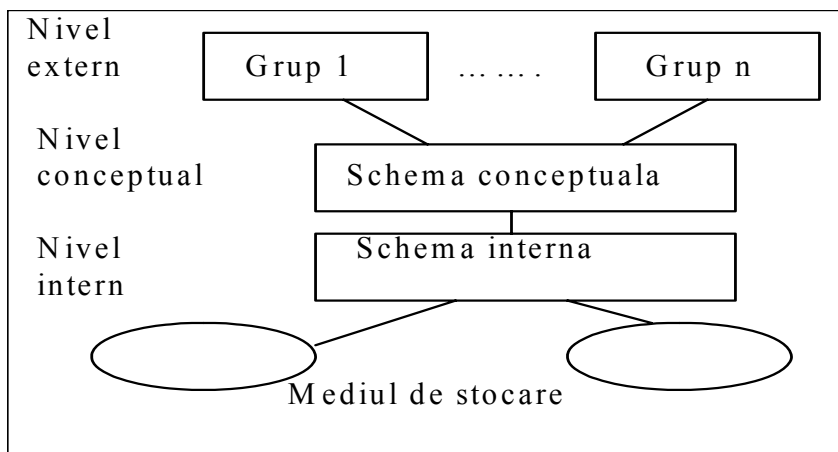


Fig. 1.1 Niveluri de abstractizare a datelor

- Nivelul extern

Este ultimul nivel de abstractizare la care poate fi descrisă o bază de date. Structurile de la nivelul conceptual sunt relativ simple, însă volumul lor poate fi deconcertant. Dacă la nivelul conceptual baza de date este abordată în ansamblul ei, în practică un utilizator sau un grup de utilizatori lucrează numai cu o porțiune specifică a bazei, în funcție de departamentul în care își desfășoară activitatea și ce atribuții au. Simplificarea interacțiunii utilizatori – bază de date, precum și creșterea securității bazei de date sunt deziderate ale unui nivel superior de abstractizare, care este nivelul extern. Astfel, structura bazei de date se prezintă sub diferite machete, referite uneori și *ca sub-scheme*, scheme externe sau imagini (view-uri), în funcție de nevoile fiecărui utilizator sau grup de utilizatori.

Observații:

Este importantă această organizare pe trei niveluri pentru că explică conceptul de independență a datelor, prin posibilitatea de modificare a sistemului bazei de date la orice nivel fără a avea influență la nivelele superioare. Independența datelor se poate defini în două moduri, ce sunt aferente nivelelor conceptual și intern.

Prin **independența logică** se înțelege capacitatea schimbării schemei conceptuale, fără a atrage după sine schimbări în schema externă sau în programele de aplicații. Este posibilă schimbarea schemei conceptuale prin expandarea bazei de date ca urmare a adăugării de noi tipuri de înregistrări sau a datelor însăși, sau prin reducerea bazei de date ca urmare a reducerii înregistrărilor.

Independența fizică este reprezentată prin capacitatea de schimbare a schemei interne fără schimbarea schemei conceptuale sau externe. Schimbarea schemei conceptuale poate surveni ca urmare a reorganizării fizice a unor fișiere, prin crearea de noi structuri de acces menite să asigure accesul eficient la date.

Accesul utilizatorului la informațiile din baza de date este posibil numai prin intermediul sistemului de gestiune a bazei de date (SGBD).

1.3 Componente ale bazelor de date relaționale

1. Tabele

Unitatea primară de stocare a datelor într-o bază de date relațională este **tabelul**, care este o structură bidimensională compusă din rânduri și coloane. Fiecare tabel reprezintă o *entitate*, ceea ce înseamnă o persoană, un loc, un lucru sau un eveniment care trebuie să fie reprezentat în baza de date, cum ar fi un client, un cont bancar sau o tranzacție bancară. Fiecare rând al tabelului reprezintă o apariție a entității.

2. Relații

Relațiile reprezintă asocierile dintre tabelele bazelor de date relaționale. Deși fiecare tabel relațional poate exista independent, esența bazelor de date este tocmai stocarea informațiilor între care există legături. De exemplu, pe lângă filmele propriu-zise, se pot stoca și informații despre categoriile folosite de magazin pentru organizarea inventarelor de filme. În același timp, puteți stoca și informații despre copiile fiecărui film, inclusiv data la care a fost primită copia și formatul acesteia, cum ar DVD sau VHS. Prin folosirea relațiilor, se pot asocia tabelele înrudite, într-un mod formal, ușor de folosit astfel încât să combinăm date din tabele multiple în aceeași interogare a bazei de date, dar păstrând flexibilitatea de a include numai informațiile care îl interesează pe utilizator. Posibilitatea de a selecta

din baza de date numai informațiile care ne interesează ne permite să ajustăm informațiile din baza de date în funcție de cerințele specifice ale persoanelor sau aplicațiilor care au acces la baza de date.

Figura 1.2 prezintă patru tabele din baza de date a magazinului de produse video și relațiile dintre acestea, într-un format cunoscut sub numele de diagramă de relații a entităților (ERD - Entity Relationship Diagram). Diagramele ERD ne pun la dispoziție o modalitate de prezentare a proiectului general al unei baze de date relaționale, într-un format ușor de înțeles pentru utilizatorii bazei de date, indiferent dacă au sau nu cunoștințe tehnice. Fiecare dreptunghi din diagramă reprezintă un tabel relațional, cu numele tabelului scris deasupra liniei orizontale și coloanele tabelului enumerate pe verticală, în porțiunea principală a dreptunghiului.

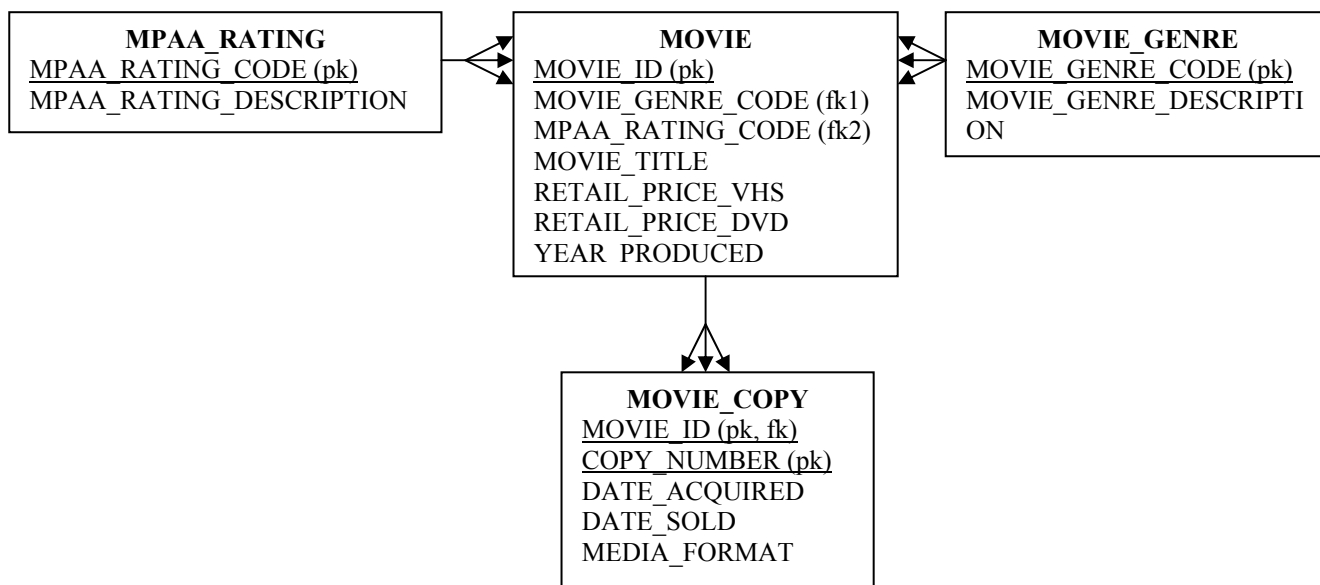
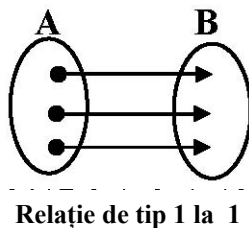


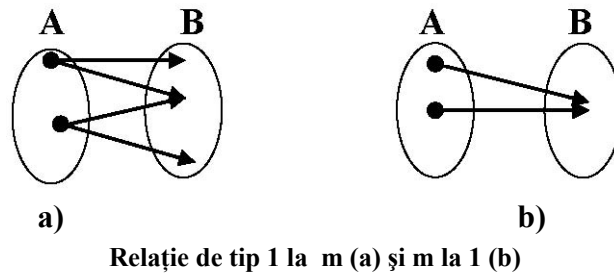
Fig. 1.2 Diagrama ERD a bazei de date pentru magazinul de produse video (prezentare parțială)

În funcție de numărul de elemente, între care se stabilesc relații, aparținând celor două colecții, aceste relații pot fi de tip unu la unu, unu la mulți și mulți la mulți.

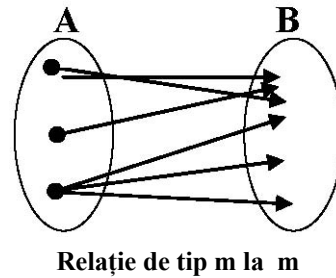
Relațiile de tipul 1→1 (unu la unu), care presupun că unui membru din colecția A îi corespunde un singur membru din colecția B.



Relațiile de tipul 1→m sau m→1 (unu la mulți sau mulți la unu), care presupun că unui membru din prima entitate A îi corespund mai mulți membri din a doua entitate B; astfel de relații se mai numesc și relații ierarhice.



Relațiile de tipul $m \rightarrow m$ (mulți la mulți), în care unui membru din entitatea A îi corespund mai multe date din colecția B și mai multor date din colecția A îi corespunde o singură dată din colecția B.



Relații de tip mulți la mulți se mai numesc și relații de tip rețea. O relație mulți la mulți se va descompune întotdeauna în două relații, o relație tip unu la mulți și respectiv o a doua relație de tip mulți la unu prin intermediul unei entități de legătură.

Fiecare relație este prezentată în diagrama ERD ca o linie ce conectează două tabele. Cele două capete ale liniei arată *cardinalitatea maximă* a relației, respectiv numărul maxim de rânduri dintr-un tabel care pot fi asociate cu un rând dat din tabelul aflat la celălalt capăt al relației.

Cardinalitatea maximă poate fi:

- *unu* (caz în care linia nu are nici un simbol special la capăt) sau
- *mai multe* (caz în care linia se termină cu un simbol numit *picior de cioară (crow'sfoot)* - linia se împarte la capăt în trei segmente).

În apropiere de capătul liniei se află un alt simbol, care arată *cardinalitatea minimă*, adică numărul minim de rânduri dintr-un tabel care poate fi asociat cu tabelul de la celălalt capăt al relației.

Cardinalitatea minimă poate fi *zero*, indicată printr-un cerc desenat pe linie, sau *unu*, indicată printr-o liniuță care taie linia relației.

De exemplu, relația dintre tabelele MPAA_RATING și MOVIE din figura 1-2 este o relație de tip *unu-la-mai-mulți*, ceea ce înseamnă că fiecare rând din tabelul MPAA_RATING (tabelul din partea „unu, numit și tabel *părinte*”) poate fi asociat cu mai multe rânduri din tabelul MOVIE (tabelul din partea „mai mulți” a relației, numit și tabel *copil*”), dar fiecare rând din tabelul MOVIE poate fi asociat cu un singur rând din tabelul MPAA_RATING.

Relația este logică, deoarece orice film lansat în SUA poate avea o singură categorie MPAA, dar o categorie poate fi asociată mai multor filme diferite. Este adevărat ca filmele sunt uneori „cenzurate” pentru a putea fi încadrate în diferite categorii, dar această problemă este rezolvată ușor, prin tratarea diferitelor versiuni ale aceluiași film ca și cum ar fi filme diferite, la fel cum facem atunci când un film este reluat cu alți actori. Este foarte important să ții seama de aceste lucruri, deoarece bazele de date relaționale acceptă numai relațiile de tip unu-la-mai-mulți sau unu-la-unu.

Cardinalitatea minimă arată dacă participarea la o anumită relație este obligatorie sau opțională.

Toate relațiile din fig. 1.2 sunt obligatorii în partea „unu” și opționale în partea „mai mulți”, aceasta fiind cea mai frecvent folosită formă de relație. Dacă ne uităm din nou la relația dintre tabelele MPAA_RATING și MOVIE, aceasta înseamnă că fiecare rând din tabelul MOVIE *trebuie* să aibă un rând corespondent în tabelul MPAA_RATING, dar nu este obligatoriu ca fiecare rând din tabelul MPAA_RATING să aibă asociat un rând din tabelul MOVIE. Dacă vrei să permiți ca inventarul de filme al magazinului să conțină titluri care nu au asociată o categorie MPAA, liniuța de la capătul dinspre tabelul MPAA_RATING al liniei care reprezintă relația cu tabelul MOVIE va fi înlocuită de un cerc. Deși sunt relativ frecvent întâlnite cazurile în care partea „unu” a unei relații nu este obligatorie, este foarte neobișnuit să aveți o relație în care să fie obligatorie partea „mai mulți” a relației, ceea ce ar însemna că tabelul părinte trebuie să aibă în orice moment cel puțin un „copil” în baza de date.

Relațiile sunt implementate folosind coloane corespondente din cele două tabele participante. În diagrama ERD, coloana sau coloanele subliniate din fiecare tabel, având în dreapta notația „pk”, reprezintă cheia primară (primary key), adică o coloană sau un set de coloane care identifică în mod unic fiecare rând dintr-un tabel.

Un tabel poate avea o singură cheie primară. Totuși, o cheie primară poate fi compusă din mai multe coloane, dacă aceasta este calea de formare a unei chei unice. Dacă o cheie primară este folosită într-un alt tabel pentru stabilirea unei relații, poartă numele de cheie externă.

Cheile primare și cheile externe sunt blocuri de construcție fundamentale ale modelului relațional, deoarece stabilesc relații și permit crearea legăturilor între date, atunci când este necesar. Trebuie să înțelegeți acest concept pentru a putea înțelege cum funcționează bazele de date relaționale.

3. Restricții

O restricție este o regulă specificată pentru un obiect al bazei de date (de obicei, un tabel sau o coloană), având rolul de a limita într-un mod oarecare domeniul de valori permise pentru obiectul respectiv al bazei de date

După ce sunt specificate, restricțiile sunt impuse automat de sistemul DBMS și nu pot fi ocolite decât dacă o persoană autorizată le dezactivează sau le șterge (le elimină). Fiecare restricție primește un nume unic, astfel încât să poată fi referită în mesajele de eroare și în comenzile folosite ulterior în baza de date. Este recomandabil ca proiectanții bazei de date să denumească restricțiile, deoarece numele generate automat de baza de date nu sunt foarte descriptive.

Există mai multe tipuri de restricții pentru baze de date:

- **Restricția NOT NULL.** Poate fi plasată pe o coloană pentru a împiedica folosirea valorilor nule. O valoare nulă (null) este o modalitate specială prin care sistemul RDBMS tratează valoarea unei coloane pentru a indica faptul că valoarea coloanei respective nu este cunoscută. O valoare nulă nu este același lucru cu un spațiu liber, un șir vid sau valoarea zero — este o valoare specială care nu este egală cu nimic altceva.
- **Restricția cheie primară (primary key).** Definită pe coloana (coloanele) cheie primară ale unui tabel pentru a garanta că valorile cheie primară sunt întotdeauna unice în întreg tabelul. Atunci când cheia primară este definită pe mai multe coloane, combinația valorilor acelor coloane trebuie să fie unică în tabel - o coloană care reprezintă doar o parte a cheii primare poate conține valori duplicate în tabel. Restricțiile cheie primară sunt aproape întotdeauna implementate de RDBMS prin folosirea unui index. Indexul este un tip special de obiect al bazei de date care permite efectuarea căutărilor rapide în valorile coloanei. Atunci când în tabele sunt inserate rânduri noi, sistemul RDBMS verifică automat indexul pentru a se

asigura că **pk** a noului rând nu este deja folosită în tabel și, dacă se întâmplă acest lucru, respinge cererea de inserare. Căutarea în indexuri se face mult mai repede decât căutarea în tabel; ca urmare, indexarea cheii primare este esențială pentru orice tabel, indiferent de dimensiunea acestuia, astfel încât căutarea cheilor duplicate la fiecare inserare să nu ducă la o reducere semnificativă a performanțelor. O caracteristică suplimentară a cheilor primare este faptul că nu pot fi definite decât pe coloane pentru care a fost definită și restricția NOT NULL.

- **Restricția de unicitate (*unique*)**. Definită pe o coloană sau un set de coloane care trebuie să conțină valori unice în cadrul tabelului. Ca și în cazul cheilor primare, sistemul RDBMS folosește aproape întotdeauna un index ca modalitate de impunere eficientă a restricției. Totuși, spre deosebire de cheile primare, un tabel poate avea definite mai multe restricții de unicitate, iar coloanele care participă la o restricție de unicitate pot conține (în cele mai multe sisteme RDBMS) și valori nule.
- **Restricția referențială** (numita uneori *restricție de integritate referențială*). O restricție care impune o relație între două tabele dintr-o bază de date relațională. Prin „impunere”, se înțelege că sistemul RDBMS se asigură întotdeauna, în mod automat, că fiecărei valori a cheii externe îi corespunde o valoare a cheii primare în tabelul părinte. Pe scurt, restricția referențială garantează că relația dintre cele două tabele și valorile corespondente ale cheii primare și cheii externe își păstrează logica în orice moment.
- **Restricția CHECK**. Folosește o instrucțiune logică simplă (scrisă în SQL) pentru a valida valoarea unei coloane. Rezultatul instrucțiunii trebuie să fie o valoare logică de adevărat (true) sau fals (false), astfel încât un rezultat adevărat să permită inserarea în tabel a valorii coloanei, iar un rezultat fals să ducă la rejectarea valorii coloanei, cu mesajul de eroare corespunzător.

4. Vizualizări

O vizualizare (view) este o interogare stocată în baza de date care pune la dispoziția utilizatorului un subset personalizat al datelor din unul sau mai multe tabele ale bazei de date. Cu alte cuvinte, o vizualizare este un tabel virtual, deoarece arată ca un tabel și, în cele mai multe privințe, se comportă ca un tabel, dar nu stochează date (nu este stocată decât interogarea SQL care definește vizualizarea). Vizualizările au mai multe funcții utile:

- Maschează coloanele pe care utilizatorul nu este nevoie să le vadă (sau nu-i este permis să le vadă).
- Maschează rândurile pe care utilizatorul nu este nevoie să le vadă (sau nu-i este permis să le vadă).
- Maschează operațiile complexe efectuate în baza de date, cum ar fi uniunile de tabele (respectiv combinarea coloanelor din tabele multiple într-o singură interogare a bazei de date).
- Îmbunătățesc performanțele interogărilor (în unele sisteme RDBMS precum Microsoft SQL Server).

1.4 Proiectarea bazelor de date relaționale. Etape. Normalizarea bazelor de date

Proiectarea unei baze de date este o activitate laborioasă și necesită parcurgerea următoarelor etape:

- formularea problemei;
- analiza cerințelor informaționale și definirea datelor de ieșire și a datelor de intrare;
- definirea tabelor, a structurii acestora și a relațiilor dintre tablele;
- optimizarea structurii bazei de date.

Odată ce acest proces a fost finalizat se continuă cu:

- proiectarea procedurilor tehnologice, pentru prelucrarea bazei de date;
- elaborarea programelor;
- testarea programelor;
- definitivarea documentației.

Toate aceste activități necesită, pentru proiectele reale complexe, o muncă în echipă pe baza unei metodologii riguroase, cunoscută ca metodologia de analiză și proiectare a sistemelor informatice. În cadrul unui sistem informatic baza de date reprezintă elementul central în jurul căruia se concentrează celelalte componente ale sistemului.

Formularea problemei presupune stabilirea obiectivelor aplicației informatice care asigură actualizarea și exploatarea bazei de date în concordanță cu cerințele managementului activității economice pentru care este proiectată baza de date. Obiectivele unei aplicații informatice sunt legate de asigurarea informațională a desfășurării proceselor decizionale specifice actului de conducere. Deci, noi trebuie să ne gândim că, prin existența unei baze de date, să asigurăm fondul de informații, într-o structură și de o calitate corespunzătoare cu cerințele managementului firmei. Baza de date trebuie să permită atât obținerea unor informații de detaliu, elementare, cât și calculul și prezentarea unor indicatori sintetici, agregați. Dacă am lua doar două obiective: reducerea costurilor și creșterea productivității muncii într-o firmă, atunci o bază de date trebuie să furnizeze informații despre consumul factorilor de producție, costurile medii și globale, despre personalul muncitor și producția realizată, despre cheltuielile salariale etc. Aceste informații vor servi conducerii la identificarea căilor de reducere a costurilor și adoptarea celor mai adecvate măsuri pentru reducerea acestor costuri. După aplicarea măsurilor în practică, informațiile stocate în baza de date trebuie să permită de data aceasta și o analiză comparată a costurilor noi cu cele vechi, de exemplu, o analiză a dinamicii costurilor pe baza indicilor statistici. Am ales acest mic exemplu didactic pentru a accentua încă o dată complexitatea procesului de proiectare a bazei de date.

Analiza cerințelor informaționale, pornind de la obiectivele formulate anterior, se concentrează asupra a două probleme:

- indicatorii, rapoartele, listele și datele de ieșire care trebuie obținute;
- datele de intrare necesare pentru obținerea datelor de ieșire.

Acestea sunt cerințele informaționale care înglobează atât cerințele pentru datele de intrare pe baza cărora se creează și se actualizează baza de date, cât și cerințele pentru datele de ieșire folosite pentru urmărirea, controlul și dirijarea activității economice. Datele de intrare se culeg, de regulă, din documentele primare care circulă în cadrul fluxului informațional al firmei. Datele finale se vor integra în ansamblul de rapoarte, liste, situații cu rezultate pe care le furnizează sistemul informațional

compartimentelor de conducere. Pentru indicatorii incluși în rapoartele finale, în general pentru oricare din indicatorii de ieșire, trebuie să fie foarte clar modul în care sunt obținuți prin prelucrarea datelor de intrare. În consecință, acolo unde este cazul, se precizează algoritmi de calcul, regulile de totalizare, sau alte reguli de obținere a fiecărei coloane, sau totaluri din rapoartele finale.

Aceasta are ca punct de plecare inventarierea câmpurilor prezente în situațiile finale și apoi gruparea lor în tabele. Gruparea câmpurilor pe tabele se realizează prin diverse metode. Dintre aceste metode, două sunt cele mai utilizate:

- analiza concordanței IEȘIRI – INTRĂRI;
- analiza semnificației semantice a datelor.

Analiza concordanței IEȘIRI – INTRĂRI este o tehnică specifică proiectării sistemelor informatice care identifică documentele primare din care se preiau datele de intrare folosite în calculul datelor de ieșire. Aceste documente vor constitui sursele de creare și actualizare a tabelelor bazei de date.

Tehnica este utilă în cazul în care proiectantul bazei de date este familiarizat cu sistemul informațional existent.

Un indicator prezent într-o situație finală se poate obține astfel:

- prin preluarea directă din documentul primar, respectiv din tabelul bazei de date;
- prin aplicarea unui algoritm de calcul.

Tabelele se compun prin gruparea câmpurilor pe principiul apartenenței acestora la anumite documente primare care circulă în cadrul sistemului informațional.

Analiza semantică se practică atunci când proiectantul bazei de date nu are la dispoziție un set de documente primare și apare în cazul sistemelor informaționale care se proiectează pentru firmele noi.

Definirea tabelor și relațiilor dintre tabele este etapa următoare în proiectarea bazei de date. Analiza cerințelor informaționale și a proceselor de prelucrare va conduce la identificarea datelor ce vor trebui stocate și care vor alcătui tabelele bazei de date. O tabelă va păstra datele fie despre toate caracteristicile unei colecții de date, fie numai pentru o parte dintre aceste caracteristici. Aici intervine spiritul analitic al proiectantului. Structura unei tabele este reprezentată de lista câmpurilor asociate tablei împreună cu descrierea atributelor fiecărui câmp (natură, lungime, număr de zecimale etc.). În structura unui tabel se regăsesc următoarele categorii de câmpuri:

- câmpuri de identificare (chei primare și chei condiționate);
- câmpuri tip dată calendaristică;
- câmpuri cantitativ-valorice;
- câmpuri de legătură cu alte tabele;
- câmpuri de stare care păstrează informații privind ultimele operații de prelucrare care au fost efectuate pe înregistrările din tabel.

Relațiile dintre tabele se caracterizează prin plasarea unor câmpuri comune în structura fiecăruia dintre tabelele aflate în relație directă. Pe baza acestor câmpuri, chei, fiecare sistem de gestiune a bazelor de date își construiește un mecanism propriu de accesare a înregistrărilor de date. Aceste mecanisme sunt transparente pentru utilizatorul obișnuit. Totuși este bine de reținut că nu orice câmp poate fi folosit la stabilirea unei relații, a unei legături între două tabele. Numai câmpurile de tip cheie candidat, care au proprietatea de a identifica în mod unic o înregistrare dintr-o tabelă, pot fi folosite în acest scop. Cheile candidate se mai numesc și indecși. Operația prin care se construiește sistemul de legături pentru ordonarea în vederea regăsirii înregistrărilor într-o tabelă se numește indexare. Prin indexare, fiecărei table principale de date i se va asocia o tabelă index corespunzătoare cheilor de

indexare.

Optimizarea structurii bazei de date este un proces prin care se urmărește:

- reducerea redundanței datelor;
- eliminarea anomaliilor de actualizare.

Reducerea redundanței datelor până la un nivel minim și controlat urmărește eliminarea duplicării inutile a unor câmpuri în mai multe tabele sau eliminarea câmpurilor obținute prin calcul pe baza câmpurilor atomice. Un anumit nivel de redundanță, însă, trebuie admis pentru a nu denatura realitatea reflectată de date. De exemplu, câmpul VALOAREA_CONTRACTULUI, se calculează după relația: VALOAREA_CONTRACTULUI=CANTITATE*PRET

Nu este recomandată eliminarea acestui câmp pe considerentul că el se obține automat prin calcul.

De exemplu, în cazul în care după un anumit interval de timp, prețurile suportă o majorare globală, cum se practică foarte des, atunci automat se vor modifica și valorile contractelor încheiate anterior datei de majorare a prețurilor ori acest lucru nu este corect, contractul odată perfectat nu-și poate modifica prețul convenit prin negociere.

Anomaliile de actualizare se referă la anomaliile de ștergere, respectiv de modificare. De exemplu, se consideră o firmă care derulează lunar mii de contracte de aprovizionare, pentru un nomenclator foarte mare de produse agroalimentare, dar care operează numai cu câțiva furnizori. Dacă în tabela CONTRACTE sunt incluse alături de codul furnizorului și denumirea furnizorului, contul său bancar și denumirea băncii, atunci va apărea următoarea anomalie de actualizare, în cazul schimbării băncii și a contului bancar de către furnizor. Acest lucru necesită parcurgerea tuturor înregistrărilor în care există aceste valori și actualizarea acestora (figura 1.4).

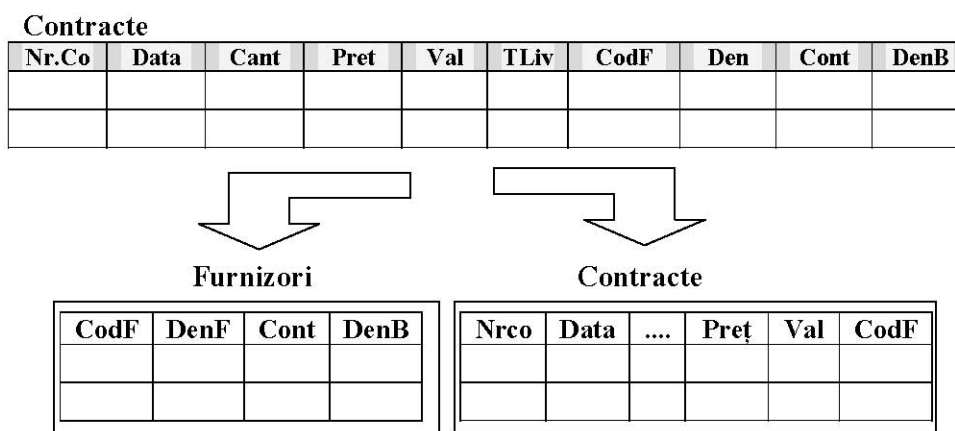


Fig. 1.4 Eliminarea anomaliilor de actualizare

Soluția este de a crea o tabelă separat, care are în structură: codul furnizorului, denumirea, contul și banca acestuia, în tabela CONTRACTE păstrându-se doar codul furnizorului ca element de legătură, ceea ce previne pierderea de informații prin spargerea unui tabel în două. În acest fel modificarea se realizează numai asupra unei singure înregistrări.

În 1972, Dr. E. F. Codd, părintele bazelor de date relaționale, și-a dat seama că tabelele relaționale care îndeplinesc anumite criterii pun mai puține probleme la inserarea, actualizarea sau ștergerea datelor. Ca urmare, a pus la punct un set de reguli care trebuie respectate (organizate în trei „forme normale”) și un proces numit normalizare, care este o tehnică pentru producerea unui set de relații (termenul folosit de Dr. Codd pentru tabele) cu proprietățile dorite.

Necesitatea normalizării

Figura 1.5 prezintă tabelul **MOVIE** fără normalizare, așa cum ar arăta dacă toate informațiile despre filme ar fi colectate într-un singur tabel. Acest exemplu va fi folosit pentru ilustrarea procesului de normalizare.

În general, numele coloanelor din tabelele relaționale folosesc liniuțe de subliniere pentru separarea cuvintelor. În discuția despre normalizare am eliminat aceste liniuțe din figuri, pentru a face textul mai ușor de citit.

Există trei probleme care pot apărea în tabelele fără normalizare din bazele de date relaționale. Scopul procesului de normalizare este de a elimina aceste probleme (anomalii) din proiectul bazei de date. Acestea sunt:

- Anomalia de inserare
- Anomalia de ștergere
- Anomalia de actualizare

Anomalia de inserare

Anomalia de inserare se referă la o situație în care nu puteți insera date în baza de date din cauza unei dependențe artificiale dintre coloanele unui tabel.

Anomalia de ștergere

Anomalia de ștergere este inversul anomaliei de inserare. Se referă la situația în care ștergerea unor date duce la pierderea neintenționată a altor date.

Anomalia de actualizare

Anomalia de actualizare se referă la o situație în care actualizarea unei singure valori necesită actualizarea mai multor rânduri. Un alt pericol legat de această anomalie este faptul că stocarea unor date redundante poate duce la posibilitatea de a actualiza numai o parte a copiilor respectivelor date, ceea ce ar avea ca rezultat apariția inconsecvențelor în baza de date.

Aplicarea procesului de normalizare

De obicei, normalizarea începe de la mijloacele de redare a datelor care sunt (sau vor fi) prezentate utilizatorilor, cum ar fi pagini web, ecrane ale aplicațiilor, rapoarte și așa mai departe. Colectiv, acestea sunt numite *vizualizări de utilizator (user views)*. Poate părea ciudat la prima vedere, dar este ceva obișnuit ca proiectarea unui sistem de prelucrare a datelor să înceapă de la rezultatele pe care le va vedea utilizatorul, parcurgând apoi drumul înapoi către mijloacele folosite pentru obținerea rezultatelor dorite.

În timpul proiectării bazei de date, procesul de normalizare este aplicat fiecărei vizualizări, iar rezultatul este un set de relații normalizate care pot fi apoi direct implementate ca tabele ale bazei de date relaționale. Procesul în sine este destul de simplu, iar regulile nu sunt foarte dificile. Totuși, stăpânirea procesului de normalizare cere timp și exercițiu, în special deoarece impune proiectantului să se gândească într-un mod conceptual la datele și relațiile pe care intenționează să le folosească.

În timpul normalizării, considerați că fiecare vizualizare este o relație. Cu alte cuvinte, conceptualizați fiecare vizualizare ca și cum ar fi deja un tabel bidimensional - un mod de lucru pentru care aveți nevoie de experiență.

Rețineți că scopul procesului de normalizare este eliminarea anomaliilor de inserare, actualizare și ștergere. Procesul determină crearea unui număr mai mare de relații decât ați avea într-un model fără

normalizare. Relațiile suplimentare sunt necesare pentru eliminarea anomaliilor, dar împărțirea datelor în mai multe relații face ca extragerea datelor stocate să fie puțin mai dificilă. De fapt, sacrificiați o parte din performanțele de extragere a datelor și din ușurința utilizării pentru ca operațiile de inserare, actualizare și ștergere să fie mai simple.

Alegerea unui identificator unic

Primul pas al procesului de normalizare constă în alegerea unui *identificator unic* (*unique identifier*), care este un atribut (o coloană) sau un set de atribute care identifică în mod unic fiecare rând de date dintr-o relație.

Identificatorul unic va deveni ulterior cheia primară a tabelului creat din relația normalizată. Pentru normalizare, *este obligatoriu ca* fiecare relație să aibă un identificator unic, în multe cazuri, puteți găsi un atribut care identifică în mod unic datele din fiecare rând al relației pe care vreți să o normalizați. Atunci când nu puteți găsi un singur atribut care să poată fi folosit ca identificator unic, este posibil să găsiți mai multe atribute care pot fi concatenate (combinat) pentru a forma un identificator unic. Atunci când identificatoarele unice sunt formate din atribute multiple, fiecare atribut rămâne pe propria lui coloană - nu faceți decât să definiți un identificator unic format din mai multe coloane.

În foarte puține cazuri, într-o relație nu există un set rezonabil de atribute care să poată fi folosit ca identificator unic. Atunci când se întâmplă acest lucru, trebuie să inventați un identificator unic, deseori cu valori atribuite secvențial sau aleatoriu pe măsură ce noile rânduri de date sunt adăugate în tabelul bazei de date. Această tehnică este sursa unor identificatoare unice, precum numărul de asigurări sociale folosit în Statele Unite, numerele de identificare ale angajaților, numerele de înmatriculare ale mașinilor sau CNP.

Prima formă normală: eliminarea datelor repetate

O relație este în *prima formă normală* atunci când nu conține atribute cu valori multiple (atribute multivaloare), adică atribute care au mai multe valori pentru același rând de date. Într-o relație, orice intersecție a unui rând cu o coloană trebuie să conțină *cel mult* o valoare pentru ca relația să fie în prima formă normală.

Pentru transformarea relațiilor ne-normalizate în prima formă normală, trebuie mutate atributele multivaloare și grupurile repetitive în noi relații.

Procedura de mutare a unui atribut multivaloare sau a unui grup repetitiv într-o nouă relație constă în următoarele etape:

1. Creați o nouă relație, cu un nume sugestiv. Deseori, este bine să includeți numele relației originale, parțial sau în întregime, în numele noii relații.

2. Copiați identificatorul unic din prima relație în noua relație. Datele depind de acest identificator în relația originală, așa că trebuie să depindă de aceeași cheie și în noua relație. Identificatorul copiat va deveni cheie externă în noua relație.

3. Mutați grupul repetitiv sau atributul multivaloare în noua relație.

4. Formați un identificator unic în noua relație, adăugând atribute la identificatorul unic copiat din relația originală. Ca întotdeauna, asigurați-vă ca identificatorul unic nou format conține numai numărul minim de atribute necesar pentru a-1 face unic. Dacă mutați un atribut multivaloare, care, în esență, este un grup repetitiv cu un singur atribut, este adăugat atributul respectiv pentru formarea identificatorului unic. Poate părea ciudat la prima vedere, dar identificatorul unic copiat din relația originală nu este doar o cheie externă, ci, de obicei, și o parte a identificatorului unic (cheia primară) a

noii relații. Acest lucru este absolut normal. De asemenea, este perfect acceptabil să avem o relație în care toate atributele fac parte din identificatorul unic (adică nu există atribute care să nu facă parte din cheie).

5. Opțional, puteți să înlocuiți cheia primară cu un singur atribut surogat pentru cheie. Dacă faceți acest lucru, trebuie să păstrați și atributele care compun cheia primară naturală, formată la pașii 2 și 4.

A doua formă normală: eliminarea dependențelor parțiale

Înainte de a explora a doua formă normală, trebuie să înțelegem conceptul de *dependență funcțională*.

Pentru această definiție, vom folosi două atribute arbitrare, inteligent denumite „A” și „B”. Atributul B este *dependent funcțional* de atributul A dacă în nici un moment nu există mai mult de o valoare a atributului B asociată cu o valoare dată a atributului A.

Se spune că o relație este în *a doua formă normală* dacă îndeplinește următoarele criterii:

- Relația este în prima formă normală.
- Toate atributele non-cheie sunt dependente funcțional de identificatorul unic (cheia primară), *luat ca întreg*.

În esență, amestecăm atribute care descriu în aceeași relație două lucruri (entități) diferite (deși înrudite) din lumea reală. Nici nu e de mirare că am obținut un asemenea haos. A doua formă normală ne va ajuta să rezolvăm problemele.

A doua formă normală se aplică numai relațiilor care au identificatoare unice concatenate (adică formate din atribute multiple). Într-o relație care are un singur atribut ca identificator unic, este imposibil ca un alt atribut să depindă de o parte a identificatorului unic, deoarece acesta, fiind format dintr-un singur atribut, nu are părți componente. **Ca urmare, orice relație în prima formă normală care are cheia primară formată dintr-un singur atribut este automat în a doua formă normală.**

A treia formă normală: eliminarea dependențelor tranzitive

Pentru a înțelege a treia formă normală, trebuie să înțelegem mai întâi conceptul de dependență tranzitivă.

Despre un atribut care depinde de un atribut care nu este identificator unic (cheie primară) a relației se spune că *este dependent tranzitiv*.

Se spune că o relație este în *a treia formă normală* dacă îndeplinește următoarele două criterii:

- Relația este în a doua formă normală.
- Nu există dependențe tranzitive (cu alte cuvinte, toate atributele non-cheie depind *numai* de identificatorul unic).

Pentru a aduce la a treia formă normală o relație aflată în a doua formă normală, mutăm atributele dependente tranzitiv în relații în care depind numai de cheia primară

Avem grijă să lăsăm atributul de care depind acestea în relația originală, cu rolul de cheie externă. Va trebui apoi să reconstruim vizualizarea originală printr-o uniune. Ca efect secundar, toate atributele ușor de calculat sunt eliminate ca încălcări ale criteriilor celei de-a treia forme normale.

De exemplu, într-o bază de date pentru vânzări, Suma Totală este obținută înmulțind Cantitatea Cumpărată cu Prețul Unitar; așa cum se observă cu ușurință, Suma Totală este dependentă de Cantitatea Cumpărată și de Prețul Unitar. Presupunând că toate cele trei atribute sunt dependente de identificatorul unic al relației care le conține, este ușor de văzut că Suma Totală (rezultatul calculat)

este, de fapt, *dependentă tranzitiv* de celelalte două atribute.

II. SISTEME INFORMATICE INTEGRATE

2.1. Definiții

În condițiile actuale ale globalizării afacerilor, mediul organizațional al unei firme trebuie să se adapteze cerințelor concurențiale ale pieței. Creșterea economică a unei firme depinde în mod esențial de abilitatea ei de a actualiza și integra, personaliza și extinde aplicațiile informatice, într-un mod flexibil și rapid, oferind tuturor utilizatorilor acces instantaneu, interactiv și consistent la modelul său de date. De asemenea, pentru asigurarea eficienței activității lor, firmele trebuie să standardizeze gestiunea proceselor economice.

Se afirmă că integrarea completă este un obiectiv major al gestiunii resurselor informaționale, care devin din ce în ce mai complexe și mai numeroase și de aceea este necesar să se realizeze și să se implementeze sisteme informatice integrate.

În cele ce urmează vom prezenta câteva accepțiuni ale acestui concept.

O primă accepțiune a noțiunii de sistem informatic integrat este dată în Hotărârea de Guvern nr. 841/1997, unde prin sistem informatic integrat se înțelege un sistem informatic care îndeplinește următoarele condiții:

- ❖ utilizează o bază de date unică;
- ❖ are în componență programe informatice, care cuprind activitățile tuturor compartimentelor funcționale ale firmei, conform organigramei acesteia;
- ❖ există un plan de securitate al sistemului informatic, care cuprinde măsuri tehnice și organizatorice corespunzătoare.

Pentru realizarea acestor obiective, se derulează diverse proiecte de integrare informațională. Derularea unui asemenea proiect nu este o activitate deloc simplă, ci reprezintă o adevărată „provocare tehnologică”, atât pentru proiectanți, cât și pentru firme.

De-a lungul existenței sale, o firmă achiziționează sau dezvoltă prin forțe proprii mai multe aplicații informatice, menite să-i satisfacă cerințele, legate de diversificarea ori extinderea activităților sale. Fiecare dintre aceste aplicații răspunde unei probleme concrete sau acoperă un anumit proces economic, fără să țină seama de lanțul de procese sau de legăturile cu celelalte aplicații informatice implementate în respectiva firmă.

În faza de început a informatizării activităților, mai ales din considerente de ordin financiar, firmele decid în general să achiziționeze sau să dezvolte prin forțe proprii o serie de aplicații informatice pentru activitățile de contabilitate, apoi pentru cele financiare, de salarizare a personalului, aprovizionări etc., fără nici o legătură între aceste aplicații.

În etapa a II-a se încearcă construirea unor legături între aceste aplicații, legături concretizate în interfețe personalizate, care își propun realizarea integrării între două sau mai multe aplicații. Fiecare dintre aplicații folosește în continuare baza sa de date proprie, ceea ce înseamnă redundanță (aceleași date) și sursă de inconsistență (datele comune trebuie actualizate separat, ceea ce înseamnă eforturi suplimentare și posibilitatea apariției unor neconcordanțe).

Într-o etapă următoare a procesului de integrare s-a trecut la implementarea pachetelor ERP (Enterprise Resource Planning). Apărute în anii '90, ele au devenit o prezență obișnuită în marile corporații și în companiile multinaționale. A doua jumătate a ultimului deceniu din secolul 20 a

însemnat deschiderea aplicațiilor de tip ERP pentru segmentul întreprinderilor mici și mijlocii.

În perioada actuală, succesul implementării pachetelor ERP în IMM-uri depinde și de măsura în care ele permit integrarea altor categorii de sisteme, cum ar fi cele privind soluțiile de tip Customer Relationship Management (CRM), SupplyChain Management (SCM), Business Intelligence (BI), precum și cele specifice utilizării Internet-ului.

Un alt factor mobilizator în procesul extinderii sistemelor integrate l-a reprezentat creșterea fără precedent a activităților de comerț și colaborare electronică (e-commerce și e-business).

Definiția și evoluția integrării

Integrarea este o activitate ce reunește oameni, echipamente, programe, dar și practici manageriale. Integrarea aplicațiilor este o abordare strategică de a lega mai multe sisteme informatice, la nivel de informații și servicii, astfel încât sistemele sunt capabile să facă interschimb de informație și să asigure o funcționare a proceselor în timp real .

Integrarea aplicațiilor software în cadrul unei întreprinderi sau între mai multe întreprinderi care colaborează este un subiect de mare actualitate. Integrarea aplicațiilor software de întreprindere permite coordonarea și sincronizarea mai multor aplicații eterogene, atât în interiorul (integrarea aplicațiilor la nivel de companie), cât și în afara întreprinderilor (integrarea aplicațiilor Business-to-Business - B2B).

Denumită în limbajul de specialitate EAI (Enterprise Application Integration), integrarea aplicațiilor la nivel de companie reprezintă, de fapt, noul stil de lucru în domeniul software. Întreprinderile au din ce în ce mai puțini informaticieni care concep și scriu aplicații și din ce în ce mai mulți care integrează aplicații. Entitatea ce trebuie integrată nu mai este un obiect sau o componentă software, ci este o aplicație software. Prin EAI, sistemele informatice ale întreprinderilor se mulează din ce în ce mai bine pe structura procesului de afaceri.

Complexitatea problemelor legate de infrastructura informatică crește și mai mult în cazul unei întreprinderi virtuale, formată din module (secții, departamente, birouri etc.), cu funcționalitate extrem de diversă și grad de dispersie geografică oricât de mare. Granularitatea modulelor se poate situa pe o scară foarte cuprinzătoare, depinzând în mare măsură, atât de specificul domeniului de activitate, cât și de posibilitățile de organizare ale întreprinderii respective.

În contextul actual în care informația este privită ca o resursă strategică a întreprinderii, a crescut foarte mult importanța integrării sistemelor informatice care să faciliteze utilizarea în comun a datelor și mișcarea lor în cadrul întreprinderii.

La nivelul anului 1999 s-a estimat că peste o treime din bugetul din industria IT a avut ca destinație proiectarea, realizarea și întreținerea unor soluții de integrare a sistemelor informatice. Dar, cele mai multe dintre aceste soluții au optat pentru varianta de integrare “punct la punct”, și s-au dovedit a fi mari consumatoare de resurse.

Dezvoltarea unei strategii eficiente de integrare a sistemelor informatice la nivelul întreprinderii este una dintre cele mai complexe probleme întâmpinate de managerii IT. Complexitatea acestei probleme rezidă în principal din faptul că cele mai multe dintre aplicații au fost dezvoltate fără a se urmări o arhitectură a sistemelor informatice sau o strategie de dezvoltare a acestora.

Începutul integrării în domeniul IT poate fi considerat momentul apariției circuitului integrat în 1959, care a reunit alte descoperiri cum ar fi: tranzistorii, rezistențele și capacitorii pe un singur chip de silicon. În 1965 Gordon Moore, unul din fondatorii Intel prezicea că numărul de tranzistori pe un microchip se va dubla la fiecare 18 luni. Această lege, încă este surprinzător de adevărată și acum, la peste 40 de ani de la formularea ei. Acesta poate fi motivul pentru care avem nevoie de integrare:

pentru a ne descurca în condițiile unei complexități crescute.

Principiile managementului complexității sunt: descompunerea în părți mai mici și mai ușor de manipulat, construirea unei interfețe standard pentru ca aceste părți să comunice și apoi dezvoltarea unei structuri ierarhice unde informația este din ce în ce mai abstractizată odată ce urcăm în ierarhie.

Termenul de middleware a apărut la sfârșitul anilor 80 pentru a descrie un software de management al rețelelor, dar a fost larg folosit începând cu 1995. Middleware-ul a evoluat într-un set de reguli și capabilități, care facilitau construirea de aplicații distribuite. Acest termen a fost legat de bazele de date relaționale și desemna o integrare bazată pe mesaje.

Informatizarea, dezvoltarea economică globală, specifice începutului de secol XXI, au accentuat tendința de organizare a sistemelor informaționale în modele din ce în ce mai complexe. Prin integrare crește precum am arătat complexitatea, dar și calitatea, pentru că reuniunea sistemelor presupune adăugarea de componente evolutive și emergente.

Dacă organizarea duce la integrare și integrarea duce la complexitate, aceasta din urmă determină la rândul ei diversificarea. Din punct de vedere al diversității, integrarea este efectul evoluției ciclice și progresive a unui mix de tehnologii și este sprijinită de performanțele și de expertiza profesioniștilor.

Integrarea aplicațiilor poate lua mai multe forme:

- integrarea internă a aplicațiilor – *integrarea aplicațiilor la nivel de companie*;
- integrarea externă a aplicațiilor – *integrarea aplicațiilor Business-to-Business*.

Cele două tipuri de integrări au multe elemente comune. De exemplu, întotdeauna vor exista: o transformare de tehnologie care va face diferența între semantica aplicațiilor, tehnologia de router prin care se va asigura că informația ajunge la destinația corectă și reguli de procesare pentru a defini comportamentul de integrare.

Strategia IT este necesar să fie cunoscută de toți factorii care influențează deciziile de integrare a proceselor economice cum ar fi configurarea proceselor economice, frontierele acestora și locul în care schimbarea este cel mai probabil a se produce. Înțelegerea scopurilor economice, cum ar fi strategiile de fuzionare și de achiziție sau cost și creșterea eficienței, apare ca o cheie fundamentală. Trebuie stabilită o perspectivă comună internă și externă a nucleului economic, de informație și de procese, pentru a înțelege relațiile și interfețele între unitățile economice, sau a partenerilor comerciali.

Trebuie stabilite problemele proprietății pentru aplicații, componente, infrastructura integratoare, interfețele externe etc. Și acestea pot fi unele dintre cele mai dificile sarcini și pot traversa actualele frontiere organizaționale și responsabilitatea acestora.

O tendință în evoluția integrării sistemelor este trecerea de la integrarea bazată pe informație la integrarea bazată pe servicii. Integrarea bazată pe informații oferă un mecanism ieftin de a integra aplicații deoarece, în cele mai multe cazuri, nu este nevoie ca aplicația să fie modificată. Cu toate că, acest tip de integrare oferă o soluție funcțională pentru multe domenii ale problematicii de integrare a aplicațiilor, integrarea bazată pe servicii oferă mai multă valoare pe termen lung.

Definiția și rolul sistemelor informatice integrate

Def. Sistemele informatice integrate desemnează sisteme complete, cu procese de afaceri, practici manageriale, interacțiuni organizaționale, transformări structurale și management al cunoștințelor.

Un sistem de aplicații integrat trebuie să reprezinte soluția pentru orice instituție care necesită

un sistem informatic modern, indiferent dacă acesta automatizează procesele interne din cadrul organizației, relațiile cu clienții sau pe cele cu furnizorii și partenerii. Adoptarea unor aplicații disparate pentru diferite activități ale fluxului de afaceri, poate reprezenta o soluție bună pentru moment, dar care poate genera mari probleme legate de fragmentarea informației și dezvoltarea ulterioară a sistemelor, prin încercarea de a integra soluții ulterioare.

Producătorii de software care oferă aplicații ce rulează pe multiple surse de date sau care nu acoperă toate sectoarele fluxurilor de afaceri, nu furnizează pachete de soluții integrate, ci mai degrabă colecții separate de aplicații, bune să rezolve problemele cerute de sisteme disparate, dar care nu reușesc să funcționeze împreună.

Principala problemă a falselor pachete de aplicații este fragmentarea informației, generată de sisteme disparate. Consolidarea informațiilor venite de la un mare număr de surse este laborioasă și costisitoare. O altă mare problemă este automatizarea incompletă, care nu acoperă toate procesele afacerii, rezultând sisteme discontinue, ce oferă funcțiuni analitice doar la nivel departamental, incapabile să asigure o viziune unitară asupra organizației. În aceste condiții, managerul instituției nu are la dispoziție decât piese dintr-un puzzle, care rareori se îmbină.

Pentru a face saltul calitativ de la acțiuni punctuale la procese de afaceri, organizațiile trebuie să adopte soluții integrate și colaborative, care să se adapteze strategiilor de distribuție și care să includă și funcționalități de suport decizional.

Un adevărat pachet integrat are aplicațiile proiectate de la început pentru a lucra împreună: acestea partajează același model de informații și informatizează procesele de business la nivelul întregii organizații.

Principalele avantaje pe care o suită de aplicații integrate trebuie să le ofere beneficiarilor sunt:

- reducerea costurilor pe termen lung;
- creșterea eficienței operaționale;
- returnarea rapidă a investițiilor în IT;
- migrarea mai rapidă la modele de e-business

Pentru a înțelege rolul jucat de un sistem informatic integrat în funcționarea unei organizații, este necesar să se pornească de la modelul general de organizare a unei afaceri, corespunzător majorității întreprinderilor de producție, comerț, servicii sau a unora din sectorul financiar-bancar.

Conform acestui model, orice întreprindere este constituită din următoarele zone:

a. Zona Back Office

Din punct de vedere informatic, această zonă se caracterizează prin:

- Importanța fundamentală a bazei de date, care poate fi situată centralizat pe un singur server sau poate fi repartizată fizic pe mai multe servere;
- Particularitățile aplicațiilor utilizate: o parte importantă dintre acestea reprezintă programe pentru tratarea “în bloc” a datelor (de exemplu, tratarea “în bloc” a tuturor tranzacțiilor unei zile de lucru). O altă parte a aplicațiilor realizează gestiunea tranzacțiilor și au ca scop tratarea simultană a unor cereri din partea unui număr mare de utilizatori;
- Importanța critică a prelucrărilor realizate, de care depinde întreaga activitate a întreprinderii;
- Centralizarea pe un număr redus de servere pe care rulează sisteme de operare dedicate.

Cea mai apreciată calitate a unui sistem de Back Office o reprezintă coerența și integritatea datelor. De asemenea, disponibilitatea continuă a sistemului și continuitatea serviciilor (chiar și în caz

de căderi sau defecțiuni) sunt caracteristici esențiale ale oricărei aplicații Back Office.

În cazul întreprinderilor moderne, aplicațiile Back Office garantează însăși funcționarea întreprinderii, de aceea se impune existența unui sistem de înaltă securitate a datelor, atât la nivel fizic, cât și la nivelul drepturilor de acces.

b. Zona Front Office

Aplicațiile Front Office sunt acele produse pe care întreprinderea le oferă clienților astfel încât să asigure servicii rapide. Principalele cerințe la care trebuie să răspundă o aplicație Front Office sunt:

- ❖ **Gestiunea relațiilor cu clienții** (Customer Relationship Management, CRM) – cuprinde instrumente de administrare a clienților (consultarea dosarelor clienților, culegerea de informații referitoare la operațiile efectuate de clienți pentru a le trimite spre procesare aplicațiilor Back Office), precum și instrumente de asistare a clienților (evaluarea în funcție de o serie de criterii, asistență în configurarea cererii și alte forme de asistare interactivă a clienților). Evaluarea clienților după o serie de criterii („scoring”) permite stabilirea gradului în care un client sau un proiect poate satisface sau nu condițiile prevăzute (de exemplu, pentru acordarea unui credit). Aplicațiile de asistență în configurarea cererii au ca scop propunerea, pe baza catalogului furnizorului și a răspunsurilor la un set de întrebări, variantele de ofertă cele mai adecvate la cererile exprimate de client;
- ❖ **Gestiunea agenților de vânzări** – are ca scop gestiunea agenților de vânzări sub mai multe aspecte: cota din vânzările totale, performanțele, realizarea cifrei de afaceri individuale și colective, obținerea rezultatelor consolidate etc;
- ❖ **Gestiunea clienților** – face parte din aplicațiile Front Office puse la dispoziția centrelor de asistență din teritoriu și utilizează tehnologii de integrare cu rețeaua telefonică;
- ❖ **Instrumente de asistare a deciziei** – reprezintă o consecință a prezenței celorlalte tipuri de aplicații menționate la nivelul fiecărei agenții sau centru de vânzări. Sunt avute în vedere instrumente de căutare și extragere de date, precum și aplicații SIAD (Sistem Interactiv de Asistare a Deciziei);
- ❖ **Gestiunea rețelei de agenții** – este un complement al aplicațiilor Back Office apărut datorită numărului mare de agenții și centre de vânzări ale întreprinderilor mari, mai ales multinaționale. Această rețea poate cuprinde sucursale proprii, concesionari sau alți agenți comerciali.

În ultima vreme, acestor două zone ale întreprinderii li s-au mai adăugat altele două:

c. Zona Middle Office

Este o zonă greu de definit la nivel fizic, care a primit în timp două accepțiuni diferite:

- Într-o primă accepțiune, aceasta reprezintă zona de Back Office prezentă în cadrul fiecărei agenții sau centru de vânzări, zona situată fizic în Front Office, dar îndeplinește funcții de Back Office;
- În a doua accepțiune, aceasta reprezintă componentele intermediare ale întreprinderii, cu rol de interfață între Front Office și Back Office, pentru gestiunea rețelei și transmiterea datelor către aplicațiile centrale (aflate în Back Office).

Din punct de vedere informatic, aplicațiile de tip Middle Office sunt cele care alimentează componentele menționate anterior. În condițiile dezvoltării unei arhitecturi client-server pe trei nivele (server central, servere agenție și stații de lucru), componentele Middle Office ale sistemelor informatice se pot găsi atât pe serverele de agenție, cât și pe serverul central.

d. Zona Web Office

Procesarea tranzacțiilor generate de serviciile la distanță s-a realizat multă vreme separat de

restul sistemului informatic. În prezent, tehnologia World Wide Web a introdus o nouă dimensiune a întreprinderii, numită generic Web Office, care completează Front Office și Back Office și dă posibilitatea conectării la sistemul informatic al întreprinderii din orice punct de pe glob.

Prin Web Office se integrează mai multe tipuri de aplicații:

- Aplicații interne ale întreprinderii – destinate exclusiv personalului din întreprindere, accesul din afară fiind oprit în general prin aplicații de tip firewall. Aceste aplicații pot furniza servicii multiple: coordonarea și gestiunea proiectelor, mesagerie internă, agendă de grup, diverse tipuri de urmărire la distanță a activității, videoconferință, etc. Se folosesc tehnologii intranet care presupun utilizarea tehnologiilor internet împreună cu produse de securitate care să protejeze domeniul și să blocheze accesul neautorizat din interior sau din afară;
- Aplicații accesibile partenerilor – destinate partenerilor în sens larg (furnizori, clienți, reseller-i, consultanți etc.) utilizează servere extranet și oferă servicii echivalente cu aplicațiile interne, dar destinate utilizatorilor externi ai întreprinderii;
- Aplicații accesibile publicului larg – asigură accesul public la serviciile întreprinderii, serverele internet realizând o extindere a activității întreprinderii spre stațiile de lucru ale partenerilor prin intermediul cataloagelor on-line cu imagini ale produselor, plăților securizate prin carte de credit sau portofel electronic etc.

Nivelul de securitate al componentelor intranet, extranet și intranet este diferit, dar nu trebuie neglijat pentru nici una dintre aceste trei zone. Este de dorit să se asigure protecția datelor de consultat și să se identifice în orice moment persoanele care navighează. Acest lucru poate fi realizat prin oferta de abonamente pentru accesul la informații, abonamente care pot fi gratuite sau nu, în funcție de serviciile oferite.

Sistemele informatice de gestiune sunt definite în literatura de specialitate urmărindu-se două abordări:

a) plecând de la informație și de la suportul acesteia;

b) plecând de la funcția pe care sistemul informatic de gestiune trebuie să o realizeze.

În primul caz, sistemele informatice de gestiune reprezintă ansamblul informațiilor utilizate în cadrul firmei, a mijloacelor și procedurilor de identificare, culegere, stocare și prelucrare a informațiilor.

În cea de a doua abordare a definiției sistemelor informatice de gestiune se pornește de la scopul acestora și anume oferirea informației solicitate de utilizator în forma dorită și la momentul oportun în vederea fundamentării deciziilor.

Def: *Sistemul informatic de gestiune asigură obținerea și furnizarea informației solicitate de utilizator, folosind mijloacele TI, pentru fundamentarea deciziilor privind un anumit domeniu din cadrul firmei.*

Sistemele informatice de gestiune (SIG) presupun definirea: *domeniilor de gestiune, datelor, modelelor, regulilor de gestiune.*

Domeniile de gestiune corespund fiecăreia dintre activitățile omogene desfășurate în cadrul firmei - cercetare-dezvoltare, comercială, de producție, de personal, financiar-contabilă – cu luarea în considerare a interacțiunilor dintre ele. Mai mult, abordarea acestor domenii se realizează într-o viziune ierarhică conducând la identificarea următoarelor nivele:

- **Tranzacțional** în cadrul căruia se efectuează operații elementare;

- **Operațional** unde se desfășoară operații curente, deciziile luate la acest nivel sunt curente, de rutină;
- **Tactic** corespunzând activităților de control și deciziilor pe termen scurt;
- **Strategic** caracteristic deciziilor pe termen lung și/sau care angajează global firma.

Datele reprezintă „materia primă” a oricărui sistem de gestiune. Sunt avute în vedere toate datele vehiculate și prelucrate indiferent de natura lor, caracterul lor formal sau informal sau de suporturile pe care se află.

Modelele de gestiune regroupează procedurile proprii unui domeniu. Putem exemplifica prin modelul:

- Contabil, specific domeniului financiar-contabil;
- Tehnologiei de fabricație specifică domeniului producției;
- De vânzări specific domeniului comercial.

Regulile de gestiune permit prelucrarea datelor și utilizarea informațiilor în conformitate cu obiectivele sistemului.

În cadrul unei firme cu activitate de producție și/sau comercială pot fi identificate următoarele reguli de gestiune:

- aprovizionarea se realizează când stocul efectiv scade sub stocul normat;
- evaluarea materialelor se realizează conform metodei FIFO;
- o materie primă se stochează în una sau mai multe gestiuni;
- pentru produsele de calitate a doua prețul se reduce cu 5% etc.

În cazul unei bănci, pentru sistemul informatic privind operațiunile de cont curent pot fi precizate următoarele reguli de gestiune:

- soldul minim 1000 RON ;
- plățile se efectuează în limita soldului;
- dobânzile calculate pentru conturile la vedere sunt 11% pe an;
- pot fi înregistrate maxim două persoane cu drept de semnătură.

Prin noțiunea de **domeniu** ajungem la conceptul de subsistem informatic de gestiune determinat pe criterii funcționale, pe care se grefează celelalte două concepte: modelul de gestiune și regulile de gestiune.

Sistemele informatice de gestiune actuale sunt sisteme integrate. Ele se caracterizează prin aplicarea principiului introducerii unice a datelor și prelucrării multiple a acestora în concordanță cu nevoile informaționale specifice fiecărui utilizator.

De exemplu, sistemul informatic integrat al contabilității se caracterizează printr-o introducere unică a datelor, preluate din documentele primare care actualizează o bază de date unică a contabilității care va fi ulterior exploatată pentru asigurarea, atât a lucrărilor specifice contabilității financiare, cât și a celor specifice contabilității de gestiune, răspunzându-se astfel cerințelor de prelucrare ale tuturor utilizatorilor.

Sistemele informatice din lumea reală sunt de fapt combinații integrate a mai multor tipuri de sisteme informatice. Acestea sunt sisteme informatice bazate pe computere care combină activitățile desfășurate de mai multe tipuri de sisteme informatice. Cele mai multe sisteme informatice sunt elaborate pentru a produce informații și pentru a sprijini luarea deciziilor la diferite niveluri ale managementului, dar și pentru ținerea de diverse evidențe și prelucrare a tranzacțiilor.

2.2 ERP – definiție, arhitectură, funcții

Un ERP, „considerat expresia cea mai fidelă a interdependenței dintre economic și tehnologia informațională, reprezintă o infrastructură software, multi-modulară ce oferă suport de gestiune și coordonare a diferitelor structuri și procese din companie, în vederea realizării obiectivelor de afaceri”¹.

Scopul ERP – sistem de gestiune integrată a proceselor de afaceri – este realizarea unei mai bune comunicări în companie, îmbunătățirea cooperării și interacțiunii dintre diferite departamente precum cele de planificare a producției, achiziții, producție, vânzări și relații cu clienții. Pe scurt, un sistem informatic de gestiune a companiei de tip ERP reprezintă planificarea celor 4 factori determinanți pentru o afacere de succes: factorul uman, financiar, tehnic și de resurse (cei 4 M - *Man, Money, Machines și Materials*)².

Davenport T.H., specialist de renume în domeniile de management și sisteme informaționale pentru afaceri propune ca definiție pentru ERP: „un pachet care promite integrarea completă a tuturor informațiilor din cadrul unei organizații [...]”³(figura 1.5.).

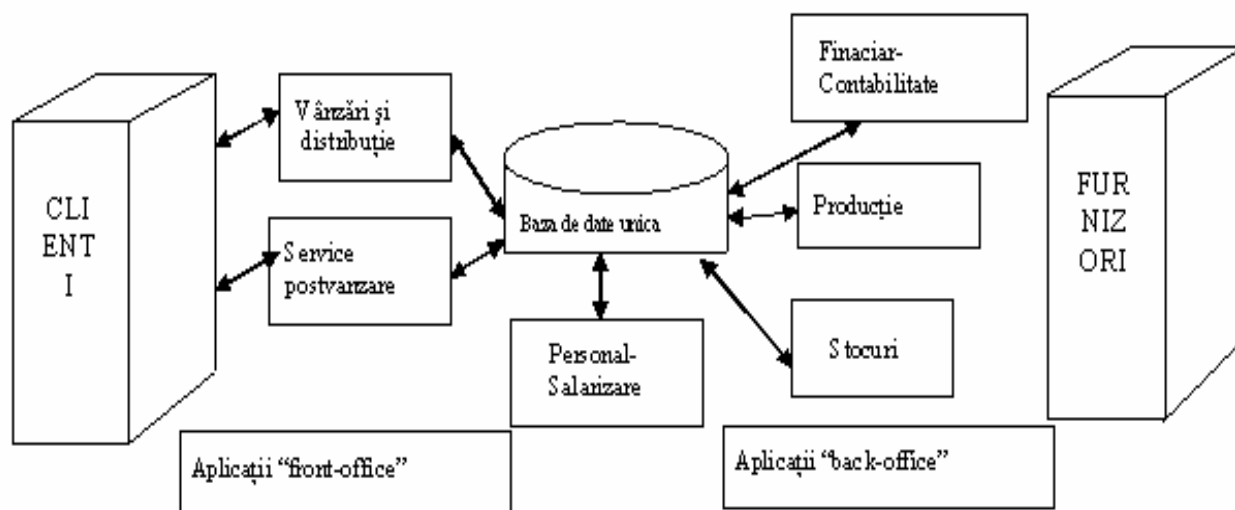


Fig. 1.5 Schema conceptuală a unui sistem ERP

ERP integrează toate procesele economice: producție, distribuție, contabilitate, financiar, personal, stocuri, service și mentenanță, logistică, gestiune de proiecte, oferind accesabilitate, vizibilitate și consistență informațională în întreaga organizație. ERP înseamnă integrarea tuturor aplicațiilor într-o soluție globală, acoperind toate procesele intercorelate ce concretizează activitatea organizației, eliminând granițele dintre departamente și delimitările funcționale, ca și pe cele ale organizației cu mediul și oferind posibilități de lucru multiutilizator, multiscop și multspațiu.

Def. Un sistem de tip ERP reprezintă o soluție software complexă, bazată pe arhitectura client-server ale cărei elemente sunt integrate într-o platformă comună, pentru gestionarea resurselor companiei, prelucrarea tranzacțiilor și facilitarea integrării tuturor proceselor necesare în cadrul unei afaceri, centralizându-le, facilitând împărtășirea datelor și eliminând redundanța. Fiecare pachet ERP oferă funcționalități diferite pentru industrii diferite.

¹ Doina Fotache, Luminița Hurbean –, „Soluții informatice integrate pentru gestiunea afacerilor-ERP”-Cap. 1, pag 10

² <http://www.cio.com/research/erp/>

³ Doina Fotache, Luminița Hurbean –, „Soluții informatice integrate pentru gestiunea afacerilor-ERP”-Cap. 1, pag. 18

Provocarea principală constă în integrarea tuturor proceselor economice și optimizarea resurselor disponibile.

Sistemele ERP actuale realizează integrarea tuturor funcțiilor de conducere ale unei companii, plecând de la:

- planificare;
- asigurarea stocului de materii prime și materiale;
- definirea tehnologiilor;
- coordonarea proceselor de producție;
- gestiunea financiar-contabilă, a resurselor umane, a stocurilor de produse finite;
- dezvoltarea și menținerea relațiilor cu clienții și partenerii de afaceri.

Un astfel de sistem permite factorilor de decizie realizarea unor analize complete asupra îndeplinirii planului de afaceri. Prin opțiunile de simulare a activităților și prin caracterul flexibil și dinamic al aplicațiilor se pot realiza:

- planuri de previziune;
- evaluări și predefiniri ale tendințelor de evoluție ale industriei din care face parte compania;
- analize calitative;
- integrarea cu noile tehnologii e-business;
- comunicare on-line.

La implementare, sistemele ERP includ o serie de caracteristici de bază. Sunt instalate pe un **sistem de gestiune a bazelor de date**. Platformele de baze de date folosite în general sunt: Oracle, DB2, Informix, Microsoft SQL Server, SQL Base, PostgreSQL, Sybase, etc. Baza de date necesită o setare inițială conform proceselor organizației și trebuie să asigure acces direct la informații în timp real (avantajul bazelor de date unice) pentru toți membrii organizației. Odată terminată instalarea, utilizatorii introduc datele, informațiile fiind transferate prin intermediul proceselor la alte module. În final, sistemele ERP includ instrumente de raportare periodice sau realizate ad-hoc.

Aplicațiile ERP sunt realizate cu ajutorul **instrumentelor CASE**, care simplifică munca programatorilor, preluând regulile și generând automat codul sursă.

Avantajele sunt: reducerea timpului de dezvoltare și obținerea unui produs de calitate, prin minimizarea erorilor. În plus, utilizarea instrumentelor CASE sprijină consistența aplicațiilor și standardizarea sub aspect funcțional. Sub o formă simplificată am putea defini ERP-ul prin prisma a doua proprietăți fundamentale: **funcționalitatea și integrarea**.

Cele două părți se intercondiționează reciproc.

Integrarea asigură conectivitatea între fluxurile de procese economice funcționale. Ea poate fi gândită ca o tehnică de comunicare. Câteva modalități obișnuite prin care comunicarea are loc, prin și pentru integrare, sunt: codul sursă, rețele locale și extinse de calculatoare, internet, e-mail, workflow, instrumente de configurare automată, protocoale, baze de date. Putem spune că integrarea este realizată prin comunicare, iar comunicarea este realizată prin integrare.

Partea funcțională a unui sistem ERP asigură fluxurile de procese economice din cadrul fiecărei funcțiuni. Astfel, în cadrul unei suite ERP se regăsesc de la câteva, până la zeci de module funcționale (contabilitate generală, debitori, salarii, stocuri, aprovizionare, planificarea producției, logistică, comenzi și vânzare)⁴.

⁴ Doina Fotache, Luminița Hurbean – „Soluții informatice integrate pentru gestiunea afacerilor-ERP „-Capitolul 1, pag.22.

Arhitectura unui sistem ERP

Sistemul aplicațiilor de întreprindere se implementează pe o **arhitectura de tip client-server** care creează premisele unui mediu de prelucrare descentralizat. Modelul de arhitectură implementat de către sistemele ERP este cel cu trei straturi, ilustrat în figura 1.6.

Caracterizarea funcțiilor celor trei niveluri ale arhitecturii:

Nivelul prezentare – constă în interfața grafică utilizator sau programul de navigare (browser) pentru accesarea funcțiilor sistemului.

Nivelul aplicație – cuprinde regulile afacerii, logica și funcțiile sistemului, programele care asigură transferul datelor de / la serverele de baze de date.

Nivelul bazei de date – asigură gestiunea datelor organizației, inclusiv a metadatelor; cel mai adesea se regăsește aici un SGBD relațional dintre cele standardizate industrial, care include și modulul SQL. Platformele de baze de date folosite în general sunt: Oracle, DB2, Informix, Microsoft SQL Server, SQL Base, PostgreSQL, Sybase, etc. Această structurare logică permite ca interfața sistemului ERP să ruleze pe calculatorul utilizatorului, prelucrarea să se realizeze pe nivelul de mijloc al serverelor de aplicații, iar sistemele de baze de date să funcționeze pe al treilea strat, al serverelor specializate.

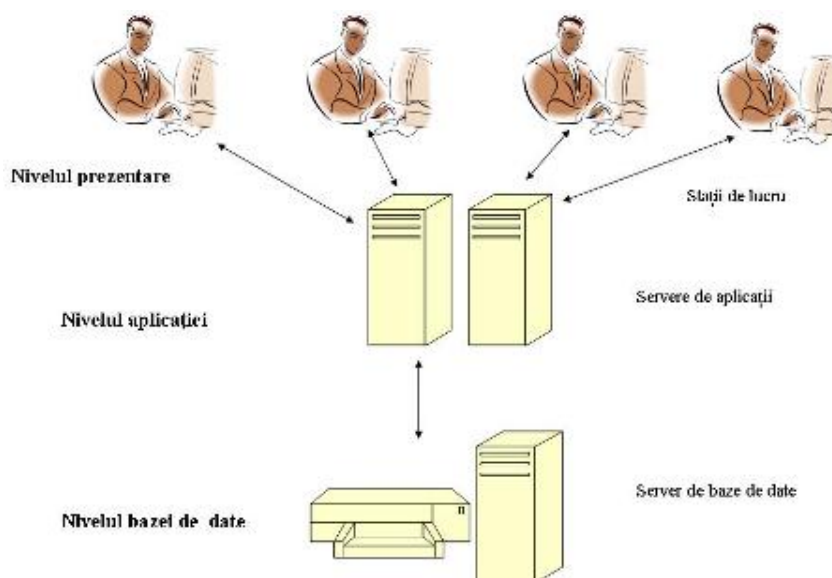


Fig. 1.6 Arhitectura cu trei straturi a unui sistem ERP

Componentele principale ale unui sistem ERP

- ✚ **Nomenclatoare (fișiere de bază)** de clienți, furnizori, personal, sub forma unor fișiere care reunesc toate datele de descriere a acestora și interfațează cu oricare modul care se servește de aceste date;
- ✚ **Contabilitate generală numită și componenta financiar-contabilă** pentru că asigură conducerea evidenței contabile și gestiunea financiară. Funcționalitățile vizează: automatizarea înregistrării informațiilor financiar-contabile preluate din documentele primare, cu preluarea automată a datelor din alte aplicații ale sistemului ERP și realizarea evidenței contabile complete, la nivel sintetic și analitic. De cele mai multe ori componenta acoperă doar cerințele contabilității financiare, asigurând în primul rând obținerea documentelor contabile de sinteză cerute de legislația în vigoare și poate fi completată printr-o componentă de analiză, tip tablou de bord, care oferă informații privind performanțele firmei;

- ✚ **Încasări-plăți.** Această componentă poate apărea sub forma a două module: *Debitori* și *Creditori*, care gestionează și înregistrează creanțele și datoriile întreprinderii;
- ✚ **Salarizare.** Componentă legată adesea de cea de resurse umane, având ca obiect calculul și evidența salariilor. Sunt automatizate calculul taxelor, al contribuției la bugetul statului și asigurărilor sociale;
- ✚ **Resurse umane.** Componenta care sprijină crearea unei politici de personal, susținând activitățile de recrutare și selecție a personalului;
- ✚ **Imobilizări.** Gestionează mijloacele fixe, dar și obiectele de inventar sau activele necorporale. Gestiunea acoperă întreaga durată de utilizare a activului și se poate afla în orice moment care este starea acestuia și operațiile efectuate asupra lui (intrare, modernizare, modificare, reevaluare, scoatere din funcțiune, casare). Oferă multiple posibilități de calcul și înregistrare a amortizării (liniară, degresivă, accelerată). Deosebit de utile sunt rapoartele generate, impuse de legislația în vigoare sau necesare conducerii;
- ✚ **Planificare-producție.** Planificarea vizează executantul, termenul, articole de realizat, costul programat și detaliile tehnice;
- ✚ **Urmărire producție** (uneori livrat într-o singură componentă împreună cu *Planificarea*) - înregistrează preluarea notelor de predare și a rapoartelor de lucru, analizează și compară comenzile lansate, oferă rapoarte cumulate ori detaliate ale producției, pe faze sau pe produse/lucrări, ca și rapoarte de costuri;
- ✚ **Gestiune date tehnice** - stochează definițiile și caracteristicile tehnice ale produselor și tehnologiilor de fabricație;
- ✚ **Planificare necesar de materiale** - determină automat cantitățile necesare, pe baza datelor despre procesul de fabricație și a planului de producție aprobat;
- ✚ **Planificare și urmărire consumuri și costuri** întocmește bonurile de consum și preia datele despre consumuri de la magazine, centralizează aceste date pentru calculul costurilor, generează rapoarte detaliate sau centralizate cu privire la consumurile planificate și realizate;
- ✚ **Managementul proiectelor** - are ca obiect proiectele de investiții, activitățile interne sau lucrările efectuate de terți: planificarea (bugetarea), finanțarea și urmărirea executării acestora;
- ✚ **Stocuri** - Gestiunea cantitativă și calitativă a stocurilor și generarea automată a documentelor contabile;
- ✚ **Gestiunea depozitelor** (inclusă adesea în modulul de *Stocuri*) - definește din punct de vedere organizatoric unitățile de stocare: tipurile de inventar și subinventar, depozite, magazine, locații, modul de localizare al stocurilor;
- ✚ **Aprovizionare (Furnizori)** - Componenta depășește atribuțiile unei aplicații de gestiune, fiind un instrument de optimizare a aprovizionării, care poate determina realizarea de economii. Modulul *Aprovizionare* se leagă de componenta *Stocuri*.
- ✚ **Vânzări** - gestionează activitățile specifice procesului de vânzare;
- ✚ **Întreținerea echipamentelor (mentenanța)** - rezolvă Gestiunea tehnică și de utilizare a echipamentelor, permite planificarea resurselor și costul lucrărilor. Foarte important este istoricul activităților de întreținere și reparații;
- ✚ **Transport (Logistica)** - permite planificarea și gestionarea activităților logistice din procesele de vânzare și distribuție;
- ✚ **Service/Servicii** - urmărește garanțiile și serviciile postvânzare;

- ✚ **Analiza (Business Intelligence)** - Modulul preia datele din baza de date, realizează diferite analize și furnizează informațiile în forma dorită de utilizator. Cele mai puternice opțiuni sunt analizele multi-dimensionale (OLAP), simulările, scenariile și prognozele;
- ✚ **Soluții specifice fiecărei industrii**
- ✚ **Generatorul de rapoarte** - utilizatorii obțin rapoartele dorite în cadrul fiecărui modul funcțional folosind datele din baza de date a sistemului ERP.

Este unanim acceptată ideea că, deși tehnologia este esențială în realizarea unui ERP, definiția acestuia trebuie să reliefeze ariile funcționale bazate pe activitățile principale ale unei firme, cum sunt: contabilitatea, producția, vânzarea, aprovizionarea, stocurile, personalul etc.

Alte definiții consideră sistemele ERP ca fiind o soluție soft, o metodă sau un pachet de aplicații, astfel:

- soluție software completă și atotcuprinzătoare pentru o întreprindere;
- metodă pentru planificarea eficientă și controlul tuturor resurselor necesare pentru prelucrarea, realizarea, expedierea și contabilizarea comenzilor clienților, în firmele de producție, desfacere ori servicii (American Production and Inventory Control Society)
- un pachet de aplicații care permite integrarea completă a tuturor informațiilor din cadrul unei organizații.

Sintetizând aceste definiții, putem desprinde următoarea concluzie, unanim acceptată de cei citați – sistemele ERP constau din module software, care acoperă toate ariile funcționale ale unei firme: marketing-ul și vânzările, service-ul, proiectarea și dezvoltarea de produse, producția și controlul stocurilor, aprovizionarea, distribuția, resursele umane, finanțele și contabilitatea, precum și serviciile informatice.

În procesul de abordare structural-tehnologică a sistemelor ERP, am considerat importantă, pentru început, prezentarea câtorva dintre caracteristicile funcționale ale unui astfel de sistem.

Examinând această arhitectură, putem remarca câteva dintre caracteristicile funcționale ale unui astfel de sistem, și anume:

- oferă informațiile necesare conducerii firmei prin intermediul bazei de date, unde se stochează tranzacțiile zilnice;
- asigură prelucrarea corespunzătoare a datelor, pe baza unor programe adecvate;
- permite utilizarea în comun a datelor din baza de date, de către toate modulele care folosesc aceste date;
- permite realizarea fiecărui tip de prelucrare (culegere date, stocare, actualizare, interogare), în mod separat;
- asigură principiul integrării sistemului, prin intermediul bazei de date unice;
- permit accesul la date în timp real;
- oferă suport multivalută și multilingv;
- sunt adaptate specificului activităților organizațiilor (activități în diferite ramuri industriale, servicii, comerț, bănci, sănătate etc.);
- permit realizarea unor adaptări fără intervenția programatorilor.

Modulele de aplicații sunt grupate în subsisteme (suite) cum sunt: financiar, producție, distribuție etc.

Avantajele utilizării ERP

A. Analiza din punct de vedere al funcționalităților oferite

După cum precizează autorul Daniel E. O'Leary în lucrarea „Enterprise Resource Planning Systems“, principalele avantaje ale folosirii unui ERP în cadrul unei companii sunt:

- ❖ Informația este introdusă în sistem o singură dată într-o bază de date foarte complexă;
- ❖ Obligă la folosirea „celor mai bune practici” din industrie;
- ❖ Permite personalizări;
- ❖ Funcționează pe o structură fiabilă de fișiere;
- ❖ Furnizează funcționalități pentru interacțiunea cu alte module;
- ❖ Furnizează instrumente pentru interogări și rapoarte ad-hoc.

Sistemele ERP furnizează informații pentru conducere și analize pentru organizații, iar cele **5 beneficii majore ale sistemelor ERP** sunt:

- informații on-line/în timp real pentru toate ariile funcționale ale unei organizații;
- standardizarea datelor și acuratețe la nivel de întreprindere;
- aplicațiile includ cele mai bune practici din industria respectivă;
- eficiența pe care o înregistrează compania;
- analizele și rapoartele ce pot fi folosite la planificări pe termen lung.

Există cinci motive majore pentru care companiile doresc să preia ERP-ul⁵:

1. Integrarea informațiilor financiare

În timp ce CEO-ul încearcă să înțeleagă performanța globală a companiei, poate găsi diferite versiuni ale realității. Departamentul Financiar are un set de valori reprezentând venitul, departamentul Vânzări un cu totul altul și alte diferite departamente pot avea fiecare o versiune diferită despre propria contribuție la venitul total al companiei. ERP-ul creează o singură versiune a realității pe care nimeni nu o poate contesta pentru că fiecare folosește același sistem.

2. Integrarea informațiilor corespunzătoare comenzii clientului

Sistemul ERP poate deveni locul unde comanda clientului este înregistrată, trimisă de CSR (Customer Service Representative) la departamentul financiar și comercial, generând emiterea unei facturi corespunzătoare care va ajunge în final la CSR. Având această informație într-un singur sistem, în loc să o cauți în diferite alte sisteme care nu pot comunica unul cu altul, companiile pot ține evidența și stadiul unei comenzi mult mai ușor, putând coordona producția, inventarul și direcționarea informației către mai multe departamente în același timp.

3. Standardizarea și eficientizarea procesului de producție.

Companiile de producție găsesc de multe ori că diferite departamente ale companiei ajung la rezultate asemănătoare folosind diferite sisteme computerizate și metode. Sistemele ERP implică un sistem care conține metode standardizate pentru automatizarea unora dintre pașii efectuați în procesul de producție. Standardizarea acestor procese și folosirea unui singur sistem integrat poate reduce timp ineficient, mări productivitatea și reducerea erorii umane.

4. Reducerea inventarului.

ERP-ul ajută ca desfășurarea unui proces de producție să se facă mult mai ușor. Acest lucru poate duce la diminuarea inventarului corespunzător personalului folosit în procesul de producție (inventar al muncii în proces), și poate ajuta utilizatorii sistemului să planifice mai bine îndeplinirea

⁵ http://www.cio.com/research/erp/edit/erpbasics.html#erp_abc

comenzilor, reducând inventarul bunurilor care au trecut prin întreg stadiul producției, aflându-se în cadrul stocurilor existente. Pentru a îmbunătăți cu adevărat flexibilitatea rețelei de aprovizionare și distribuire, este nevoie de un software “rețea”, iar ERP-ul ajută la acest lucru.

1. **Standardizarea informațiilor din departamentul de Resurse Umane**

Mai ales în afacerile cu diferite departamente, Resurse Umane poate să nu aibă o metodă simplă pentru găsirea angajaților și pentru a comunica cu ei despre servicii și beneficii. Sistemul ERP poate îmbunătăți acest lucru.

Producția este cel mai important proces în lanțul valorii într-o întreprindere producătoare, iar calitatea și competitivitatea pe piață a produselor rezultate din procesul de producție este esențială. Pentru îndeplinirea acestor deziderate este esențială eficiența sistemului informatic de gestiune a activității. Numai implementarea unei soluții informatice perfect modelate pe specificul activităților unei întreprinderi producătoare poate asigura premisele competitivității acesteia.

Avantajul cel mai important al unui sistem informatic integrat (ERP) constă în gestionarea în mod unic a tuturor categoriilor de date și a informațiilor specifice beneficiarului.

B. Analiza din punct de vedere al costurilor și riscurilor implicate

Foarte mulți beneficiari se plâng de depășirea bugetelor inițiale aprobate pentru achiziționarea soluției ERP. Dacă implementarea are loc fără întârzieri care să presupună alocarea de noi resurse umane și materiale, atunci, de cele mai multe ori, putem vorbi despre costuri ascunse („hidden fees”).

Sorin Dimofte (SIVECO România) este de părere că apariția acestor „**hidden fees**” este cauzată de lipsa concordanței dintre poziția de la negociere și cea din timpul implementării. Cu alte cuvinte, la negociere, cerințele clientului sunt minime cu scopul scăderii prețului de achiziție, însă în timpul implementării solicitările sunt maxime, în discordanță cu negocierea inițială. Ca atare, vom vorbi și despre costuri de analiză a afacerii respective, despre costurile de personalizare a aplicației, despre licența modulelor, despre cerințele tehnice ale aplicației (licențe de utilizatori, investiții în hardware) și despre întreținerea acesteia (armonizarea cu legislația locală, suport tehnic, update-uri). Toate aceste costuri alcătuiesc TCO (Total Cost of Ownership), care nu ar trebui să fie ascuns de către furnizorii de ERP, ci, din contră, ar trebui expus încă de la primele discuții.

În al doilea rând, este vorba de **costul licențelor**, respectiv al serviciilor de configurare impuse de software-ul adițional (rețea, bază de date etc.) pe care furnizorul de ERP nu le declară «din prima» pentru că el consideră că nu fac parte din costul ERP. Nu sunt declarate nici necesitățile de upgrade al hardware-ului necesar pentru a rula soluția, față de care furnizorul consideră că echipamentele nu sunt problema lui.

În al treilea rând, pe parcursul implementării, odată cu înțelegerea mai adâncă a aplicației beneficiarului, i se revelează alte rezolvări sau necesități care atrag costuri suplimentare.

Astfel, soluțiile ERP costă între 400.000 euro și 300 milioane euro (conform unui studiu Meta Group), în funcție de:

- mărimea firmei;
- specificul de activitate;
- gradul de dispersare geografică;
- infrastructura tehnologică ⇒ ERP este scump pentru orice tip de afacere.

Investițiile sunt de multe ori adevărate „bombe cu ceas” pentru bugetele companiilor. Totuși, firmele care au implementat pachete Enterprise Resource Planning (ERP) trebuie să recunoască faptul că unele cheltuieli sunt supraevaluate, în timp ce altele sunt subestimate.

Principalii „devoratori” de buget în implementarea pachetelor ERP sunt :

✚ Training-ul

Training-ul este o componentă neglijată în procesul de implementare, atât ca etapă, cât și la nivel bugetar. Astfel, costurile sunt subestimate cel mai adesea pentru că se pierde din vedere faptul că majoritatea angajaților au de învățat un nou set de procese și nu doar o noua interfață software.

✚ Integrarea și testarea

Testarea compatibilității pachetelor ERP cu alte programe software este o altă cheltuială de care nu se prea ține seama în procesul de implementare. O companie standard de producție poate avea aplicații add-on pentru partea de logistică, taxe, planificarea producției și codurile de bare. Dacă la toată această listă se adaugă și customizarea funcțiilor de bază din pachetul ERP, costurile de integrare, testare și mentenanță a sistemului vor arunca în aer bugetul. În ceea ce privește training-ul, testarea integrării ERP trebuie făcută din perspectiva procesului.

✚ Conversia datelor

Transferul datelor, de tipul informațiilor referitoare la clienți și furnizori, sau la designul produsului, de pe vechile sisteme pe cele ERP, este o operațiune costisitoare. Deși puțini CEO sunt dispuși să admită că multe din datele existente în sistemele primite moștenire nu sunt foarte importante. Companiile neagă de multe ori redundanța datelor pe care le dețin, cel puțin până în momentul în care trebuie să le facă transferul în noile setări client/server necesare pachetelor ERP. Și din nou companiile subestimează costurile, de această dată pe cele necesare transferului.

✚ Analiza datelor

De multe ori datele din sistemele ERP trebuie combinate cu datele din sistemele externe pentru a putea permite realizarea de analize. Utilizatorii care efectuează analize în mod curent ar trebui să aibă în vedere în bugetul alocat implementării ERP-ului și depozitul de date, dar și efortul necesar pentru a-l pune pe picioare. Utilizatorii sunt puși în dificultate în acest moment: reactualizarea zilnică a tuturor datelor ERP dintr-un imens depozit de date, dintr-o corporație este dificilă, iar sistemele ERP nu ajută prea mult la identificarea acelor date care s-au modificat de la zi la zi, făcând actualizări selective. O soluție costisitoare este "custom programming".

✚ Consultanți „ad infinitum”

Pentru a evita notele de plată kilometrice către consultanți, companiile trebuie să identifice clar obiectivele către care partenerii de consultanță trebuie să îi orienteze pe angajați. Activitatea consultanților trebuie să poată fi evaluată, iar pentru aceasta este necesar să le fie stabilit un "plafon" în activitatea lor: de exemplu, un număr de angajați să poată trece cu brio un test de project management.

✚ Specialiștii

Cei mai mulți analiști susțin că succesul unui sistem ERP depinde de echipa care lucrează la implementarea lui. Software-ul este prea complex și schimbările din zona de business prea dramatice pentru a lăsa un astfel de proiect pe mâna unor amatori. Vestea proastă este că, la finalul proiectului, companiile trebuie să fie pregătite să înlocuiască mulți dintre specialiști.

✚ Echipele de implementare – „o poveste fără de sfârșit”

Multe companii tind să trateze implementările de ERP ca pe orice alt proiect software. Odată ce programul software a fost instalat, gândesc ei, echipa va fi dizolvată și fiecare își va relua postul inițial cu tot cu atribuțiile ce îi revin. După o implementare de sistem ERP însă, membrul echipei nu se poate reîntoarce la vechile atribuțiuni pentru că este mult prea valoros pentru companie - ajunge să știe mai multe lucruri despre procesul de vânzări și de producție chiar decât agentul de vânzări, respectiv agentul de producție însuși. Companiile nu își prea pot permite să trimită pe vechile posturi

participanții la proiect, întrucât există multe lucruri de făcut după instalarea ERP-ului. Doar crearea de rapoarte pentru a extrage informații din noul sistem ERP ține echipa ocupată cel puțin un an de zile. Din păcate, puține companii au în vedere etapa post-instalării și chiar și mai puține o prevăd în planificarea bugetară prealabilă implementării.

✚ În așteptarea beneficiilor

În managementul de proiecte software tradiționale, companiile se așteaptă să vadă rezultatele imediat ce instalarea aplicației s-a încheiat, lucru imposibil în cazul ERP. Majoritatea sistemelor ERP își relevă eficiența după o perioadă mai îndelungată de la implementare, iar echipa de proiect la rândul său nu primește recompensa decât după amortizarea investiției.

✚ Depresia Post-ERP

Conform unui studiu efectuat de o firmă de consultanță, una din patru companii intervievate a suferit o scădere a performanței după implementarea sistemului ERP. Motivul este acela că totul funcționează și arată altfel decât până atunci, iar când angajații nu știu exact cu ce lucrează se panichează, acest lucru afectând întregul business.

C. Factorii de risc

În privința riscurilor, se întâmplă adesea ca bugetele alocate și/sau termenele prevăzute să fie cu mult depășite – unii analiști apreciază că aproximativ jumătate din proiectele ERP nu reușesc să atingă obiectivele propuse. Cazuri celebre precum Boeing, Panasonic sau Siemens ilustrează eșecul proiectelor, în sensul ratării obiectivelor propuse ori depășirii nepermise a bugetelor.

În majoritatea cazurilor publicate eșecul implementării unui pachet de aplicații integrate s-a datorat problemelor organizaționale. Într-un top al motivelor se regăsesc:

- tratarea ERP ca pe un sistem software;
- lipsa implicării managerilor executivi (top-manageri);
- concentrarea eforturilor pe instalarea software-ului și pe „învățarea” acestuia;
- așteptări nerealiste în privința duratei de implementare;
- utilizarea sistemelor ERP pentru colectarea, prelucrarea datelor și obținerea informațiilor;
- neimplicare și neacceptare din partea utilizatorilor;
- implementări realizate de consultanți și specialiști externi;
- lipsa pregătirii psihologice corespunzătoare a utilizatorilor;
- comunicare defectuoasă între membrii echipelor de proiect;
- proiectul nu a fost pregătit corespunzător ori resursele necesare dezvoltării sale au fost insuficiente.

Având în vedere costurile destul de mari pe care le implică achiziționarea unui ERP, era firesc să întrebăm ce riscuri presupune investiția într-o asemenea soluție. Extrem de interesant este punctul de vedere al furnizorilor, aproape toți considerând că riscurile vin din partea clientului și nu din partea companiilor care se ocupă de implementare.

SAP este de părere, de exemplu, că *principalul risc ține de necunoașterea domeniului și de neglijarea anumitor criterii de alegere a soluției ERP*. Mulți manageri urmăresc, exclusiv, prețul de achiziție fără să țină seama de alți factori. De aceea există riscul de a alege o soluție nepotrivită necesităților companiei, o soluție care nu poate susține, pe termen lung, creșterea afacerii, o soluție insuficient testată care poate genera probleme de stabilitate, funcționalitate sau extindere.

Primul pas pentru o implementare de succes este alegerea unei soluții performante, apropiată de specificul de activitate al clientului astfel încât nivelul de configurare să fie minim, o soluție care să fie confirmată de referințe pe piața românească. Printr-o comunicare adecvată între cele două echipe de proiect, problemele legate de implementare pot fi evitate.

Pe lângă situațiile în care managementul companiei client nu susține proiectul sau șefii de compartimente nu se implică suficient în implementarea și utilizarea soluției sau personalul nu cunoaște și nu înțelege necesitatea și beneficiile trecerii de la aplicațiile individuale la soluțiile integrate, au fost identificate și o serie de **puncte critice** care apar pe parcursul implementării unei soluții ERP:

- în primul rând nu se acordă suficientă atenție procesului de **analiză** a nevoilor, adevăratele necesități apărând mult mai târziu, pe parcursul implementării;
- în al doilea rând, se așteaptă ca verificarea rezultatelor să se facă de către implementator, ceea ce este complet greșit, fiecare utilizator final trebuie să verifice rezultatele activității de care răspunde;
- nu în ultimul rând, tergiversarea renunțării la aplicațiile vechi, deși nu este indicat să se lucreze în paralel mai mult de două luni de zile, clienții continuă această practică până la jumătate de an.

Deosebit de important este faptul că eșuarea unei implementări nu înseamnă numai pierderea investiției materiale în soluția ERP. Efectele negative se resimt în deteriorarea relației cu clienții, cu angajații sau în pierderea cotei de piață și de imagine. De aceea se recomandă alegerea unui partener cu experiență, capabil să prevadă și să combată aceste riscuri.

Reticența clientului la sugestiile furnizorului de ERP constituie un risc deloc de neglijat. Se presupune că o companie serioasă deține un background solid pentru a putea propune practici eficiente de afaceri și modalități de îmbunătățire a mediului de lucru. Însă, în condițiile în care clientul nu este deschis la propunerile furnizorului, există riscul ca practicile greoaie utilizate până în acel moment să se perpetueze în noul sistem, scăzându-i valoarea.

Dacă echipa care realizează implementarea nu înțelege sau nu are capacitatea de a răspunde necesităților clientului, soarta ERP-ului achiziționat poate fi un eșec.

În condițiile actuale ale economiei de piață concurențiale, menținerea și întărirea poziției economice a unei societăți comerciale, indiferent de mărimea acesteia, este imposibilă fără informatizarea activității sale. Prin urmare, este imperios necesar ca întreprinderea să-și conceapă și să realizeze sau să-și achiziționeze un sistem informatic, cu ajutorul căruia să-și gestioneze evidența zilnică și să ofere managerului sprijin pentru fundamentarea deciziilor strategice și tactice, care să-i conducă activitatea și să-i consolideze poziția pe piață și eficiența sa economică.

Spre deosebire de preocupările similare cu 10-15 ani în urmă, dinamica vieții economice impune astăzi ca firmele să folosească sisteme informatice integrate, care să informatizeze toate compartimentele funcționale ale firmei. Această tendință a apărut ca o reacție firească la noile provocări ale tehnologiilor informatice moderne, într-un moment în care se constată accentuarea fenomenului de globalizare a economiei și de amplificare a concurenței între diferite firme.

Chiar și întreprinderile mici și mijlocii (IMM), antrenate în procesul de modernizare a întregii lor activități, nu pot rămâne nici ele nepăsătoare la aceste provocări majore ale tehnologiilor informatice actuale bazate, fie pe implementarea unor sisteme informatice, fie pe suportul unor sisteme distribuite. În acest context, aplicarea noilor tehnologii informatice va trebui să asigure identificarea de

noi resurse, gestionarea corectă și eficientă a patrimoniului firmei, în scopul obținerii unei eficiențe ridicate.

Analizând tabelul 1.1, se poate concluziona la prima vedere că folosirea sistemelor ERP se justifică pentru organizații de dimensiune medie/mare și mare (în special datorită consumului mare de timp și a costurilor). O analiză mai atentă scoate însă în evidență avantajele competitive esențiale: informația de calitate, dimensiunea colaborativă și deschiderea spre e-business, care le fac absolut necesare în cadrul unei economii moderne.

Tabelul 1.1

Dezavantaj	Mod de combatere/diminuare
Proiecte consumatoare de timp	Implicarea activă a managementului, obținerea consensului și acceptului general;
Costuri mari	Dificil și nerecomandat ;
Dependența de furnizor	Analiză atentă a celor două alternative: furnizor unic, sau mai mulți furnizori, prima însemnând implicarea furnizorului pe termen lung, a doua oferind șansa alegerii soluțiilor “best of breed”;
Complexitate	Selectarea doar a modulelor care sunt absolut necesare;
Necesitatea extinderii și dezvoltării ulterioare a sistemului	Poate fi eliminată, dar va reduce potențialul sistemului, care va deveni la un moment dat inefficient.

Grile

1. Serviciile pentru organizarea și întreținerea bazei de date includ:
 - a) funcții pentru salvarea și refacerea bazei de date în caz de eroare
 - b) funcții pentru normalizarea bazei de date
 - c) mecanisme de securitate pentru împiedicarea accesului neautorizat
 - d) funcții pentru formatarea textului
 - e) servicii web.

2. Componentele bazelor de date relaționale pot fi:
 - a) relații de tip 1 la 1
 - b) interogări ale bazei de date
 - c) fișiere de index
 - d) rapoarte cu datele din baza de date
 - e) reguli specificate pentru obiectele bazei de date sub formă de restricții.

3. Restricția de unicitate într-o bază de date asigură:
 - a) folosirea valorilor nule
 - b) împiedică folosirea valorilor nule
 - c) validarea valorilor unei coloane
 - d) valori unice pe o coloană sau un set de coloane ale tabelului
 - e) o relație între două tabele.

4. Funcțiile vizualizărilor sunt:
 - a) stochează datele în baza de date
 - b) ascund coloanele pe care utilizatorul nu e nevoie să le vadă
 - c) ocupă mai puțin spațiu în baza de date container
 - d) optimizează timpul de răspuns al bazei de date.

5. Optimizarea structurii bazei de date urmărește:
 - a) proiectarea procedurilor tehnologice de prelucrare a bazei de date
 - b) analiza cerințelor informaționale și definitivarea datelor
 - c) analiza semantică a datelor
 - d) scăderea redundanței datelor
 - e) eliminarea anomaliilor de actualizare.

6. Care sunt nivelurile de abstractizare a datelor:
 - a) fizic, conceptual, extern
 - b) intern, global, extern
 - c) baze de date, vederi, rapoarte
 - d) tabele libere, tabele înglobate în baze de date, vederi
 - e) formulare, grid-uri, rapoarte.

7. Câte forme poate lua integrarea aplicațiilor:
 - a) integrarea internă și cea externă
 - b) integrarea bazelor de date, a datelor și a bazei informaționale
 - c) integrarea business-to-consumer și business-to-government
 - d) integrarea aplicațiilor la nivel de companie și a aplicațiilor Business-to-Business
 - e) integrarea contabilității și a gestiunii firmei.

8. Orice întreprindere este constituită din mai multe zone din punct de vedere al modelului general de organizare a unei afaceri:
 - a) zona de rezervă și zona de producție
 - b) zona back office, front office, middle office și web office
 - c) zona de gestiune, de contabilitate, de producție și de desfacere
 - d) zona financiară și zona materială
 - e) zona executivă, zona informațională și zona productivă.

9. Care este scopul unui ERP într-o companie:
 - a) realizarea unei mai bune comunicări între diferitele compartimente ale companiei
 - b) diminuarea factorului uman ca o presiune salarială
 - c) găsirea unor soluții mai bune în ceea ce privește producția companiei
 - d) cheltuieli financiare mai mari ale companiei
 - e) venituri financiare mai mari ale companiei.

10. Modelul de arhitectură implementat de către sistemele ERP este cel cu trei straturi, care sunt acestea:
 - a) Nivelul conceptual, global, fizic

- b) Nivelul prezentare, aplicație, al bazei de date
- c) Nivelul de baze de date, formulare și rapoarte
- d) Nivelul ERP, CRM și Business-to-Business
- e) Nivelul de gestiune, al contabilității și de producție

Răspunsuri

1 – a,c; 2 – a,e; 3 – d; 4 – b; 5 – d,e; 6 – a,b; 7 – a,d; 8 – b; 9 – a; 10 – b.