

Generic Integrating Business Architecture

Lect.dr. Mureșan MIHAELA
Universitatea Creștină „Dimitrie Cantemir”, București

The generic business architecture offers an efficient solution for the business engineering and re-engineering processes. This approach strengthens the cooperation between the main actors involved in the business architecture design and implementation, aiming at including all the significant views in a integrated model. The main goal of the development of generic business architectures is to offer a standard model for the integration of the internal processes and for a better management of the technological and informational resources of the enterprise. Such standardization has as main benefits the increase of the management quality and the efficiency in the business engineering processes.

Keywords: business engineering, business architecture, business framework, model driven architecture

Dinamica în domeniul proceselor economice și al tehnologiilor de vârf impune o reconsiderare a strategiei de restructurare a proceselor de afaceri pentru o mai bună adaptare la cerințele actuale ale globalizării. În acest sens, are loc o tranziție către procesele digitale, pentru care organizația trebuie să fie pregătită atât cu infrastructura tehnologică, cât și cu cea organizațională. Complexitatea acestor schimbări a ridicat problema găsirii unor soluții care să permită o adaptabilitate și o eficiență sporite pentru asigurarea rapidă a acestor transformări. Datorită faptului că la momentul actual, trecerea la procesele digitale impune utilizarea sistemelor informatice, aceleași probleme se ridică și pentru sistemele informatice care deservesc procesele de afaceri. „Întreprinderea condusă prin sisteme informatice este o întreprindere rentabilă, reactivă și suplă. Calitatea gestiunii interne a informației devine un factor de eficacitate” [4].

Astfel, a apărut și s-a dezvoltat conceptul de arhitectură a proceselor de afaceri și de arhitectură software care redau modul de structurare a proceselor de afaceri și respectiv a sistemelor informatice care le deservesc.

Arhitectura unui sistem se bazează pe abordarea conceptuală a unui domeniu, aplicându-se succesiv operații de observare, abstractizare și generalizare. Rezultatele conceptualizării sunt formalizate pentru a fi reprezentate, ceea ce conduce la elaborarea unui model.

Arhitectura reprezintă astfel întreaga viziune integrată și formalizată asupra unui domeniu de afaceri sau asupra unei organizații în ansamblul ei. Modelarea oferă suportul de prezentare, care la rândul său se bazează pe un metamodel. Metamodelul reprezintă specificațiile unui model, principiile, semantica și sintaxa acestuia, ceea ce permite utilizarea sa în procesul de formalizare a modelului realizat prin asamblarea viziunilor asupra domeniului analizat.

În prezent, ca urmare a restructurării profunde a proceselor economice, nu există o modelare consistentă a acestora la nivelul unei întreprinderi, ceea ce induce dificultăți în abordarea sistemului informațional și respectiv informatic, tributare în același timp schimbărilor tehnologice din domeniul informației și comunicației. Pe de altă parte, sistemele informaționale, respectiv informatice exercită, prin modelele tehnice implementate o influență substanțială asupra restructurării proceselor economico-financiare. Trebuie subliniat totodată că soluțiile tehnologice de vârf, implementate punctual sub forma unor sisteme care funcționează izolat, nu contribuie la susținerea schimbării, ci dimpotrivă accentuează starea inerțială a întreprinderii față de promovarea schimbărilor necesare. În consecință, este necesară o strategie de integrare a viziunilor asupra proceselor de afaceri, a sistemului informațional și a celui informatic, pe diferite nivele de abordare: conceptual, logic,

fizic, ceea ce ar contribui la creșterea reactivității întreprinderii în raport cu schimbarea. În acest sens, având în vedere totodată complexitatea sistemului și integrarea acestuia la nivel intern și extern, rezultă necesitatea existenței unei structuri arhitecturale stabile, utilizată ca punct de referință pe parcursul dezvoltării și îmbunătățirii sistemului. Se propune astfel crearea și utilizarea unei arhitecturi generice, care să corespundă unui model general, conform viziunilor la nivel de întreprindere (proces de afaceri), sistem informațional și sistem informatic.

1. Cadrul arhitectural al întreprinderii

În 1987, John Zachman a introdus o nouă abordare, și anume a reprezentării proceselor sub forma mai multor perspective, corespunzătoare viziunilor actorilor implicați. Deși persoanele implicate, având roluri diferite, au și puncte de vedere diferite, elementele asupra cărora își formulează opinia sunt aceleași. Astfel, a fost elaborat un cadru, reprezentat sub formă matricială, în care liniile reprezintă perspectivele actorilor, iar coloanele elementele implicate în procesele de afaceri. Zachman, pornind de la scopul unei organizații de a fi producătoare de profit, identifică elementele generale din arhitectura unei întreprinderi și persoanele implicate, cu rolul particular al fiecăreia. Astfel el introduce conceptul de arhitectură la nivelul întreprinderii. Arhitectura întreprinderii, în viziunea lui Zachman, reprezintă un plan detaliat "enterprise blueprint." [8]. În acest sens, ea este un container care stochează planurile de afaceri, modelele de date și de procese și toate specificațiile și elementele proiectate la nivel conceptual, logic și fizic. În această accepțiune, cadrul arhitectural reprezintă o hartă a tuturor legăturilor posibile între inițiativele de afaceri, datele necesare pentru a sprijini aceste inițiative, procesele de afaceri care utilizează aceste date și sistemul informațional la nivel fizic, care deservește procesele și cerințele de informare. Datorită conexiunilor pe care le realizează, cadrul arhitectural devine un instrument puternic pentru managementul resurselor informaționale și al tehnologiilor. Acest cadru arhitectural oferă un suport pentru schimbare, în condițiile în care

întreprinderile se găsesc în fața unei provocări majore respectiv re-ingineria totală a proceselor, ceea ce implică noi obiective, noi procese, noi măsuri și noi tehnologii.

Oferind un tablou complet al datelor și proceselor, cadrul arhitectural Zachman permite organizației să identifice și să controleze redundanța referitoare la date și funcționalitate. Prin încorporarea legăturilor între inițiativele de afaceri, structurile informaționale și sistemele informatice, se oferă un cadru adecvat orientat pe obiective pentru re-ingineria proceselor și integrarea componentelor existente ale sistemului informațional. Descrierea comprehensivă și logică a datelor contribuie la creșterea reutilizării și portabilității structurilor de date și proceselor. Arhitectura permite, de asemenea, efectuarea analizei înainte, dar și pe parcursul implementării schimbărilor induse de modificările proceselor tehnologice și economice, pentru a se identifica efectele noilor cerințe asupra resurselor informaționale și impactul modificărilor propuse asupra sistemului actual.

Cadrul arhitectural oferit de Zachman permite "examinarea ansamblului de cunoștințe despre o întreprindere privită ca un tot" [6]. În condițiile actuale, resursele cheie ale unei întreprinderi sunt cunoștințele și datele, alături de capital, personal și resurse materiale. Deci, sistemele informaționale ca o componentă esențială a întreprinderii, interrelaționează cu procesele de afaceri. Deci, impactul schimbărilor din cadrul proceselor de afaceri induce mutații semnificative asupra colectării și procesării datelor. Altfel formulat, sistemele care oferă suport pentru schimbarea proceselor, trebuie de asemenea schimbate, lor revenindu-le și rolul de a facilita schimbările [8].

În acest fel, Zachman integrează viziunile principalilor actori implicați în cadrul întreprinderii: inițiatorul unei afaceri (inițiază și planifică o afacere într-un anumit domeniu - planificator), proprietarul (conduce afacerea - managerul), analistul de sistem (analizează și reprezintă afacerea într-o formă sistemică, ordonată), designerul (încearcă să rezolve problemele), realizatorul sistemului (construiește sistemul pentru a deveni funcțional),

constructorul/administratorul sistemului (implementează și întreține sistemul la nivel fizic). În cadrul matricei propuse de Zachman, rolurile participanților în cadrul sistemului sunt reprezentate pe linii. Corespunzător fiecărei categorii de persoane implicate, se disting viziuni diferite și respectiv nivele de abordare distincte:

- *nivel contextual* - scopul și obiectivele unui anumit domeniu de afaceri;
- *nivel conceptual* - definirea naturii afacerii, inclusiv structura, organizarea, funcțiile;
- *nivel logic* - modelul sistemului informațional care rafinează în termeni riguroși modelele definite la nivelul conceptual;
- *nivel fizic* - modelul tehnic în funcție de soluțiile tehnologice adoptate pentru implementare;
- *nivel fizic de detaliu* - specificațiile tehnice referitoare la implementarea modelului fizic pentru date, prelucrări și comunicare în contextul dat;
- *sistem funcțional* - implementarea sistemului ca parte a organizației.

Elementele cu care interacționează diferiții actori sunt reprezentate pe coloane. Fiecare coloană corespunde unei anumite dimensiuni (materiale, comportamentale, spațiale, temporale, motivaționale). Aceste elemente răspund cerințelor de bază ale unei afaceri, și anume:

- *materialele necesare* ("Ce resurse?" - identificate sub forma datelor);
- *comportamentul sistemului* ("Cum se desfășoară procesul?" - descrierea proceselor);
- *locațiile de desfășurare a proceselor* ("Unde?" - rețeaua de amplasare în spațiu);
- *încadrarea în timp a evenimentelor* ("Când?" - dinamica proceselor);
- *persoanele/organizațiile implicate* ("Cine?" - persoane fizice sau juridice care interacționează pentru derularea proceselor);
- *motivațiile și restricțiile comportamentale* ("De ce?" - factorii care influențează pozitiv sau negativ desfășurarea proceselor).

Coloanele reprezintă dimensiunile efortului de dezvoltare și prin parcurgerea fiecăreia de la prima la ultima linie, se remarcă rafinarea descrierii până la nivelul fizic, de implementare.

1. Datele. Toate liniile interferează cu datele, începând de la prima linie unde sunt prezentate datele la nivel general care afectează evoluția și scopul întreprinderii. Aceste date sunt rafinate succesiv sub formă de modele conceptuale pentru viziunea asupra afacerii, fiind apoi traduse într-un mod organizat la nivel logic din perspectiva sistemului informațional. Modelul logic se transpune la nivel fizic, ceea ce implică o descriere tehnică adecvată mediului de implementare, care la nivel de detaliu se completează cu specificațiile detaliate corespunzătoare bazei de date realizate, care asigură funcționalitatea sistemului.

2. Funcțiile. Coloana reprezentând funcțiile descrie procesul de traducere a misiunii întreprinderii în definiții tot mai detaliate ale operațiilor care asigură îndeplinirea scopului și obiectivelor organizației. La nivelul inițial, sunt descrise principalele tipuri de activități, care apoi sunt rafinate sub forma unui model conceptual al prelucrărilor, care apoi este rafinat într-un model care ilustrează transformarea datelor de intrare în date de ieșire. La nivelul modelului tehnic, se definesc modulele care redau prelucrările necesare și modul de interacțiune între acestea și de asigurare a interfeței sistemului. Codarea sau generarea automată a codului reprezintă realizarea fizică la nivel de detaliu, implementabilă sub formă de programe executabile în cadrul sistemului funcțional.

3. Rețeaua. Această coloană redă dimensiunea spațială a întreprinderii, ilustrând distribuția geografică a activităților acesteia. Descrierea devine tot mai detaliată și mai riguroasă, de la nivelul inițial, în care sunt enumerate locațiile unde întreprinderea activează, la detalierea unei arhitecturi de comunicare (interacțiunea diferitelor locații) și de rețea (distribuția datelor și a echipamentelor). La nivel fizic de detaliu se descriu adresele tuturor componentelor de rețea și protocoalele și facilitățile de comunicație.

4. Persoanele. Această coloană prezintă persoanele (fizice și juridice) care interacționează cu procesele de afaceri și cu elementele tehnologice. În primul moment interesează o identificare a unităților organizatorice și a

misiunii acestora. La nivel conceptual această listă este transpusă sub forma unor diagrame organizaționale, surprinzându-se legăturile cu funcțiile (coloana 2). Se pun totodată problemele de securitate în termeni generali care sunt ulterior rafinate. La nivel logic, interacțiunea între persoane și tehnologie se descrie în termeni mai concreți, respectiv prin descrierea necesităților fiecărei persoane care interacționează cu sistemul. Pe baza acestor descrieri se proiectează interfața între utilizator și sistem, precum și arhitectura sistemului de securitate. Acestea vor fi convertite în elemente concrete de interfață care interacționează între ele și care asigură comunicarea cu utilizatorul sub controlul sistemului de securitate.

5. *Timpul*. Coloana a cincea reprezintă dimensiunea temporală, care interacționează în strânsă legătură cu alte coloane și în special cu cea a funcțiilor (coloana 2). La nivel strategic, dinamica este redată pentru ciclurile implicate de procesele de afaceri și evenimentele semnificative pentru acestea. În cadrul modelului de afaceri sunt descrise evenimentele și condițiile care interacționează cu funcțiile sistemului. Sunt prezentate transformările de stare pe care evenimentele le produc asupra datelor. Aceste evenimente devin mesaje prin care se asigură comunicarea în cadrul proceselor interne. Toate elementele identificate sunt implementate în cadrul modulelor și programelor la nivel fizic de detaliu.

6. *Motivația*. Prin această coloană s-a dorit ilustrarea modului de transformare a obiectivelor și strategiilor de afaceri în mijloace și finalități specifice. Este descris astfel un set de restricții sau premise transformate în reguli de afaceri care se aplică asupra datelor și funcțiilor având o influență pozitivă sau negativă asupra proceselor. Diagrama de reprezentare a matricei lui Zachman denumită, *cadru arhitectural al întreprinderii* [6] este reprezentată în tabelul 1.

Acest cadru arhitectural prezintă ca principal avantaj simplitatea abordării în reprezentarea obiectelor complexe, independent de metodele și tehnicile utilizate pentru realizarea descrierii. Cadrul Zachman devine astfel un in-

strument neutru ce poate fi utilizat cu succes în planificarea și rezolvarea completă a problemelor.

Matricea prezentată poate fi utilizată ca un cadru standard de organizare la nivelul unei întreprinderi, companii de succes implementând deja acest model de arhitectură. Pe baza cadrului Zachman, compania Visible a lansat un *model universal* (Visible Advantage) care permite companiilor să dezvolte, în condiții de înaltă calitate, sisteme executive, depozite de date și sisteme de asistare a deciziilor. Utilizând filosofia de structurare a lui Zachman, se pot dezvolta șabloane generice pentru procesele de afaceri care apoi pot fi valorificate de orice întreprindere pentru a le adapta propriei sale culturi. În acest mod, sunt economisite eforturile a luni de zile de construire a arhitecturii informaționale, timp și energie care pot fi convertite în alte direcții (managementul calității etc.).

Cadrul arhitectural poate fi utilizat dezvoltarea arhitecturii întreprinderii, atât pentru ingineria înainte (forward) cât și pentru cea înapoi (reverse). Ingineria înainte include planurile de afaceri și modelarea datelor și proceselor conform proiecției asupra dezvoltării viitoare a organizației. Ingineria înapoi include analiza și documentarea tuturor structurilor existente ale organizației. Rezultatul este un model, cu documentația aferentă, ce reprezintă o viziune integrată a cadrului arhitectural al întreprinderii, în care sunt rezolvate toate redundanțele și discrepanțele și sunt optimizate circuitele și fluxurile informaționale.

În acest fel, se poate crea un instrument reutilizabil pentru laborioasa activitate de definire a modelelor arhitecturale pentru procesele de afaceri. Aceste modele se adresează proprietarilor, proiectanților și celor care dezvoltă sisteme ("back-office"), având totodată un impact direct asupra utilizatorilor ("front-office").

2. Premisele dezvoltării unor arhitecturi generice software integrate în arhitectura întreprinderii

În realizarea unui cadru arhitectural generic, s-a pornit de la premisa că toate organizațiile desfășoară un set comun de procese subordo-

Tabelul 1 - Cadrul arhitectural Zachman

	Date (Ce?)		Funcții (Cum?)		Spațiu-Rețea (Unde?)		Persoane (Cine?)		Timp (Când?)	Motivație (De ce?)
Scop/Obiective Nivel contextual - Planificator -	Listă elemente importante afaceri Clase de obiecte de afaceri		Listă operații implicate în afaceri Funcție - clase de procese de afaceri		Listă locații operaționale Noduri - locații importante afaceri		Listă unități organizaționale și persoane Persoane - organizații Activitate - misiune		Listă evenimente/cicluri semnificative Timpul - evenimente majore	Listă obiective și factori strategici Finalitate - obiective majore Mijloace - factori critici de succes
	Cerințe date	Cazuri de utilizare	Cerințe funcționale	Cazuri de utilizare	Cerințe distribuie	Cazuri de utilizare		Cazuri de utilizare		
Model de afaceri Nivel conceptual - Proprietar -	Model conceptual date Entități/ clase de obiecte - obiecte de afaceri Asocieri- relații între obiectele de afaceri		Model procese afaceri Proc. - procese de afaceri I/O - resurse de afaceri		Logistică rețea Rețea de afaceri Noduri și legături		Model flux de activități Persoană - unitate organizatorică Activitate - produs realizat		Program de activități Timp- eveniment afaceri Ciclu - ciclu de afaceri	Plan de afaceri Finalitate - obiective Mijloace - strategie
	Diagrame ER	Diagrame clase / colaborări	Diagrame flux	Diagrame activități	Diagrame de colaborări	Model de comunicație	Diagrame organizare	Roluri. Plan securitate		
Model sistem informațional Nivel logic - Designer -	Model logic al datelor Entități, clase de obiecte Asocieri - relații între entități, clase de obiecte		Arhitectură aplicație Proc - funcție a aplicației I/O - viziuni utilizatori		Arhitectură distribuită a sistemului Nod - stocare Legătură - caracteristici		Interfața cu utilizatorul Persoană - rol Activitate - rezultat funcțional		Structura de prelucrare Timp - Eveniment sistem Ciclu - ciclu de prelucrare	Modelul regulilor de afaceri Finalitate - aserțiuni structurale Mijloace - aserțiuni ale acțiunilor
	Entități Normalizare	Clase obiecte	Structură aplicație	Structurare pachete	Model geografic sistem. Roluri	Date. Acces	Diagrame flux interfață	Diagrame acces		
Model tehnologic Nivel fizic - Realizator sistem -	Model fizic de date Date - tabele, obiecte Asocieri - pointeri, chei		Proiect sistem Proc - funcție computerizată I/O - seturi de date		Arhitectură rețea Nod - hardware/ sistem software		Arhitectură prezentare Persoană - utilizator Operații - meniuri, formate ecran		Structură control Timp- execuție Ciclu - ciclul componentei	Design reguli Finalitate - condiții Mijloc - acțiune
	Descriere fizică obiecte/ entități, chei și legături		Diagrame sistem Pseudo-cod		Arhitectură hardware/ software pentru rețea		Machete meniuri, ecran, rapoarte			
Reprezentare detaliată Nivel utilizare - Constructor/ Administrator sistem -	Arhitectură stocare date Definire date		Pachet de programe Programe, proceduri Proc - instrucțiuni I/O - bloc control		Arhitectură rețea Nod - adresă Legătură - protocol		Arhitectură sistem securit Persoană - identitate Activitate - job		Definire sincronizări Timp - întrerupere Ciclu - ciclu mașină	Specificare reguli
	Atribut - câmp	Asocieri - adrese	Codare		Schemă rețea		Generare elemente de interfață. Definire resurse, utilizatori, drepturi acces			
Sistem funcțional	Date convertite		Programe executabile		Facilități de comunicare în rețea		Persoane instruite		Evenimente de afaceri	Reguli implementate

nate scopului de a realiza profit, cum sunt: marketing, distribuție, management, producție, gestiune financiară, management resurse umane etc. Deci, indiferent de profil, se poate distinge un cadru generic de procese comune care în general utilizează aceleași informații. Aceste modele generice redau procesele comune conform regulilor de afaceri și a datelor cheie vehiculate, oferind cadrul de la care se pot dezvolta soluțiile particulare pentru fiecare întreprindere. Standardizarea unor astfel de arhitecturi generice ar contribui la asigurarea transferului de experiență către utilizatori deoarece încorporează practici verificate, utilizate în procesele de control și decizie asupra arhitecturii proprii întreprinderi. În majoritatea cazurilor, nu există o arhitectură a proceselor de afaceri sau a sistemului informațional la nivelul întreprinderii, iar elaborarea acesteia reprezintă un proces laborios și costisitor. Pornind de la o arhitectură generică, utilizatorul are o viziune de ansamblu, generală, fiindu-i mult mai ușor să o particularizeze pentru propriile sale nevoi. Datorită faptului că aceste arhitecturi generice înglobează viziuni diferite, utilizarea lor nu se rezumă numai la sfera ingineriei proceselor de afaceri, ci vizează direct managementul la nivel strategic și tactic.

Abordarea actuală trebuie să conducă la integrarea arhitecturii întreprinderii cu cea a sistemului informațional și a sistemului informatic, deoarece sistemele informatice contribuie la remodelarea proceselor de afaceri și a celor informaționale, iar sfera lor tinde să se suprapună integral peste cea a sistemului informațional.

Pornind de la cadrul arhitectural Zachman și de la exemplul oferit de Visible, considerăm că existența unor arhitecturi generice ar favoriza procesele de ingineri/re-inginerie a proceselor de afaceri, respectiv a produselor software, facilitând coeziunea dintre procesele economice și implementarea la nivelul tehnologiilor informaționale și de comunicație actuale. Având în vedere tendințele actuale de trecere la economia digitală, nu se mai poate vorbi de arhitecturi ale proceselor de afaceri fără a îngloba arhitectura software ca element de bază al arhitecturii întreprinderii. Dezvol-

turea arhitecturii software integrate asigură premisele remodelării proceselor de afaceri pentru o mai bună adaptare la contextul actual și pentru creșterea competitivității în condițiile globalizării. Pentru realizarea arhitecturii software se pot utiliza tehnicile de modelare orientate obiect care oferă instrumente facile și sugestive (UML – Unified Modeling Language), asigurând accesul larg la structurile arhitecturale pentru categorii variate de utilizatori. Astfel, în matricea oferită de cadrul arhitectural Zachman se pot plasa toate tehnicile de modelare care se utilizează în procesele de inginerie software. În cadrul diagramei (tabelul 1), față de matricea originală, sunt sugerate la nivelul celulelor, soluții alternative sistemice sau orientate obiect care pot fi utilizate pe parcursul dezvoltării arhitecturii software.

Introducerea și utilizarea arhitecturilor software a creat dezbateri ample asupra acestui concept. În acest sens, există mai multe definiții dintre care unele sunt semnificative. În accepțiunea lui Barry Boehm, arhitectura software reprezintă “o colecție de software și componente ale sistemului, conexiunile dintre acestea și restricțiile aferente lor, exprimate într-o mod rațional” [1]. Această colecție abstractă poate fi utilizată oricând pentru a demonstra că elementele componente, conexiunile dintre acestea și restricțiile care definesc sistemul, dacă ar fi implementate, ar satisface cerințele actorilor implicați în sistem. O primă încercare de clarificare în domeniul definirii arhitecturii software a fost întreprinsă la prima reuniune pe probleme de arhitectură a sistemelor software [9]. Prin distilarea definițiilor și a punctelor de vedere, se distingueau, la acel moment, mai multe viziuni, și anume:

- **modele structurale**, care cuprind componentele și conexiunile dintre acestea plus alte elemente adiționale care variază în funcție de autor (stil de configurare, restricții, proprietăți, cerințe), dezvoltându-se chiar limbaje de formalizare a descrierilor arhitecturale (*ADL* - Architectural Development Language);
- **modele de tip cadru**, care sunt similare cu cele structurale, dar accentuează structura coerentă a sistemului ca întreg, fiind dezvoltate

tate pentru domenii specifice;

➤ **modele dinamice**, care se concentrează pe comportamentul calitativ al sistemului, urmărind în special modificările în configurație sau referitoare la date;

➤ **modele ale proceselor** care au ca sferă construcția arhitecturii, urmărind pașii și etapele implicate în procesul de realizare a arhitecturii.

Ulterior, Booch, Rumbaugh, și Jacobson au introdus conceptul de arhitectură software ca fiind: “un set semnificativ de decizii despre organizarea unui sistem software, selectarea elementelor structurale din care este compus sistemul și a interfețelor acestora, împreună cu comportamentul specific interacțiunii dintre elemente, compunerea elementelor structurale și comportamentale în subsisteme tot mai mari și utilizarea unui stil unitar care să ghideze organizația în tratarea elementelor și interfețelor, a structurii, a interacțiunilor și a compunerii tuturor acestora” [2].

Dintr-un alt unghi, arhitectura software este definită ca “un set de concepte și decizii de design referitoare la structura și textura software-ului, definite înainte de procesul de inginerie concurentă, pentru a permite o evaluare efectivă satisfăcătoare asupra cerințelor funcționale semnificative explicite, a celor implicite și de calitate, referitoare la o familie de produse software, o problemă sau un domeniu, exprimate într-o manieră arhitecturală” [7].

Viziunile diferite nu se exclud și nici nu reprezintă poziții antagonice asupra conceptului de arhitectură software. Ele redau spectrul comunității de cercetare în domeniul arhitecturii software și demonstrează importanța acestei paradigme, a părților constitutive și a utilizării practice a conceptelor. La o privire mai atentă, aceste definiții, deși nu exprimă în mod declarat aceeași accepțiune, pot fi tratate în ansamblu lor, observându-se că, în esență, ele rafinează elemente de bază comune, ceea ce demonstrează de fapt existența unui consens.

Datorită diversității formulărilor existente, s-a impus standardizarea terminologiei, ceea ce a condus la apariția standardului IEEE 1471. În viziunea acestui standard, conceptul de ar-

hitectură este “organizarea fundamentală a unui sistem reprezentată prin componentele acestuia, relațiile dintre ele și principiile care stau la baza conceperii și evoluției sistemului” [10].

Așa cum reiese din definițiile prezentate, arhitectura software oferă o viziune de nivel înalt asupra sistemului, punând în evidență structura acestuia, dar ascunzând detaliile de implementare. Sunt astfel asamblate perspectivele diferite generându-se o imagine sistemică asupra întregului. Prin prezentarea structurii componentelor, a relațiilor statice și dinamice între aceste componente, se realizează o proiecție a unor viziuni diferite asupra sistemului, viziuni care reprezintă abstractizări destinate unor anumite scopuri. Din cele prezentate, reiese că se remarcă cel puțin două viziuni distincte: viziunea conceptuală și logică și viziunea de implementare (tehnică). În ambele viziuni este necesară o abordare statică și/sau dinamică, precum și urmărirea dependențelor dintre ele.

Dezvoltarea unei arhitecturi generice integrate încorporează arhitectura software ca o soluție pentru managementul proceselor interne și pentru asigurarea conectivității interne și externe a întreprinderii. Rezultă astfel o arhitectură orientată model (Model Driven Architecture) care poate utiliza paradigmele obiectuale pentru realizarea unui set de modele independente de contextul de implementare.

3. Niveluri de abordare în cadrul arhitecturilor generice pentru întreprinderi

Luând ca punct de referință cadrul arhitectural Zachman, cu viziunea sa integratoare asupra proceselor de afaceri, se poate proceda la o rafinare succesivă a nivelelor, pornind de la cel contextual până la cel de utilizare. Astfel, se pot reuni, într-o viziune comună, nivelele contextual, conceptual și logic rezultând un singur nivel conceptual, iar nivelele fizic și de utilizare să se contopească într-un nivel denumit generic tehnic. În viziunea integrată a arhitecturii întreprinderii (figura 1), pentru satisfacerea necesităților diferite de utilizare, propunem deci două nivele de abordare a arhitecturii, și anume: nivelul conceptual (*arhitectură conceptuală*); nivelul tehnic (*arhitec-*

tură tehnică).

Aceste două niveluri arhitecturale includ proiecțiile viziunilor distincte asupra proceselor, obiectelor și comunicării dintre acestea ale diferiților actori implicați. Având în vedere faptul că sistemul informatic acoperă aproape integral sfera informațională, iar sfera informațională reflectă procesele de afaceri la nivelul întreprinderii, arhitectura conceptuală deservește atât interesele managementului și proprietarilor afacerii, cât și interesele analiștilor și proiectanților de sisteme informaționale și informatice. La nivelul unor arhitecturi conceptuale se identifică structura organizatorică și funcțională a sistemului. Într-o abordare obiectuală, folosind limbajul UML (Unified Modeling Language), se pot folosi cazurile de utilizare pentru descrierea generică atât a aspectelor funcționale, cât și a celor tehnice. La nivel conceptual se poate utiliza o viziune sintetică, descrisă cu ajutorul pachetelor și a legăturilor dintre acestea, viziune care se poate detalia expandându-se fiecare pachet în termeni de clase, relații și interacțiuni. Astfel se poate construi o arhitectură generică la nivel conceptual pe două nivele de detaliere. Arhitecturile generice conceptuale reprezintă un nivel care rămâne deschis tuturor actorilor ce au tangență cu sistemul.

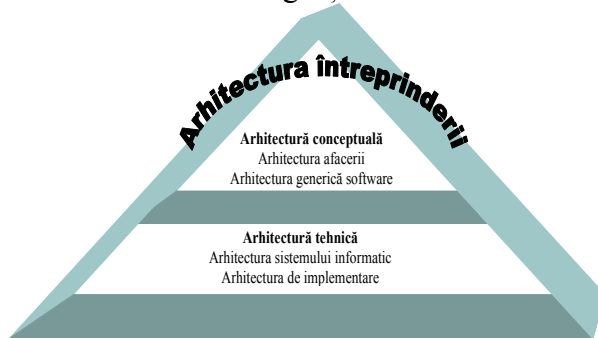


Fig.1. Arhitectura întreprinderii

La nivelul tehnic sunt identificate soluțiile de implementare a arhitecturii conceptuale. Arhitectura tehnică se dezvoltă pe baza celei conceptuale, utilizând ca modalitate de reprezentare componentele, constituite din unități de program prin care se realizează la nivel fizic funcționalitatea sistemului. La nivel tehnic, se vor construi diagramele componentelor și cele de amplasare care vizează latura de distribuire a sistemului. Nivelul tehnic este cu precădere orientat informatic și destinat

specialiștilor în domeniul tehnologiei informației

În acest mod, prin definirea unor structuri arhitecturale generice reutilizabile și modificabile conform strategiei adoptate, se facilitează concentrarea asupra obiectivelor și factorilor de succes, proiectarea și implementarea beneficiind de utilizarea componentelor cu caracter universal. Succesul unor astfel de arhitecturi generice constă în coerența acestora, astfel încât ele constituie un nucleu universal pentru un anumit domeniu. Arhitecturile generice la nivel conceptual se constituie într-o coloană vertebrală pentru structurarea proceselor de afaceri și pentru un management eficient al întreprinderii.

Noua abordare reprezintă de fapt premisele *ingineriei afacerilor*, a cărei sferă este aceea de creare a unor arhitecturi pentru procesele economico-financiare, mentenabile în condiții de dinamică accentuată a contextului intern și extern în care evoluează întreprinderile. Prin viziunea de ansamblu, ca și prin legătura între modele, arhitecturile generice devin instrumente ale managementului resurselor informaționale și tehnologice.

Bibliografie

- [1] Boehm Barry, (1995), <http://www.sei.cmu.edu/architecture/definitions.html>, 2002
- [2] Booch, Rumbaugh, Jacobson - The UML Modeling Language User Guide, Addison-Wesley, 1999
- [3] Coad Peter, Lefebvre Eric, De Luca Jeff - *Java modeling in color with UML*, Prentice Hall, 1999
- [4] Cornion J., Haltab, N., J., *Qui a encore peur de l'informatique*, Eyrolles, Paris, 1990
- [5] Fowler M., *Analysis Patterns – Reusable Object Models*, Addison Wesley, 2000, U.S.A.
- [6] Hay David C., *The Zachman framework: An introduction*
- [7] Mehdi Jazayeri, Alexander Ran, Frank van der Linden - *Software Architecture for Product Families: Principles and Practice*, Addison Wesley Longman, 2000
- [8] Alan Perkins, *Implementing the Zachman Framework for Enterprise Architecture* - <http://www.visible.com>
- [9] ***, First International Workshop on Architectures for Software Systems, Mary Shaw, 1995, <http://www.sei.cmu.edu/architecture/definitions.html>, martie 2002
- [10] ***, IEEE 1471/ 2000, Recommended Practice for Architectural Description of Software-Intensive Systems