

Comerç Electronic
(E-Commerce)

Adevăr sau ficțiune?

- VISA a procesat 5200 de tranzacții pe minut pe perioada Sărbătorilor de Iarnă?
- Ficțiune: VISA a procesat peste 5200 de tranzacții **pe secundă**
- Primul ATM a fost dezvoltat în anul 1939?
- Adevăr: Luther George Simjian a patentat Bankomatic, o versiune mai puțin funcțională a ATM-urilor moderne
- Primul card de credit universal pentru folosire în restaurante a fost emis în anul 1930?
- Ficțiune: În 1950, Frank McNamara, fondatorul Diner's Club, a inventat primul card universal de credit pentru restaurante

Sisteme Comerț Electronic (E-Commerce)

- Primele sistemele de comerț electronic au apărut în anii 1990:
 - se focalizau pe integrarea funcțiilor de comerț și utilizatori;
 - implicau un număr mare de grupuri de utilizatori, fiecare cu propriile cerințe și așteptări.
- Caracteristici actuale ale sistemelor de e-Commerce:
 - Integrează tehnologiile IT cu afacerile;
 - Trece dincolo de granițele unei organizații;
 - Adaptare la schimbare;
 - Depășirea limitelor tradiționale (geo-politice);
 - Schimbarea sensului tradițional de afacere economică.

Definiții ale conceptului de Comerț Electronic (E-Commerce) (1)

- Perspectiva on-line:
 - Cumpărarea și vânzarea de produse și servicii de către companii și consumatori folosind infrastructura Internet
- Perspectiva comunicațională:
 - Livrarea de informații, produse/servicii sau efectuarea plășilor peste linii telefonice, rețele de calculatoare sau alte mijloace electronice

Definiții ale conceptului de Comerț Electronic (E-Commerce) (2)

- Perspectiva proceselor de business:
 - Aplicarea tehnologiei electronice în scopul automatizării tranzacțiilor și fluxurilor de afaceri

- Perspectiva serviciilor de business:
 - Instrumentul ce adresează dorința firmelor, a consumatorilor și a managementului de a limita costurile simultan cu îmbunătățirea calității produselor și creșterea vitezei de livrare a serviciilor

Comerț Electronic (E-Commerce)

- Definiția cu care lucrăm: **prezentarea, vânzarea și comercializarea bunurilor și serviciilor prin intermediul tehnologiilor oferite de Internet**
- Beneficii:
 - Operabilitate 24 ore/zi, 7 zile pe săptămână
 - Costuri interne mici. Se elimină interacțiunea dintre cumpărător și intermediarii tranzacției
 - Costuri scăzute de procesare a tranzacției
 - Reducerea erorilor, procesul fiind automatizat
 - Posibilitatea de a prezenta/vinde practic orice produs sau serviciu
 - Prezență globală
 - Disponibilitatea informațiilor despre clienți

Avantajele sistemelor Comerț Electronic (E-Commerce)

- Promovarea produsului se realizează mai amplu și mai rapid;
- Dispar limitele geografice de vânzare;
- Economii la vânzarea/cumpărarea produsului;
- Timpul redus de acces la produs;
- Adaptarea la schimbări poate fi mai rapidă prin utilizarea tehnologiilor comerțului electronic;
- Îmbunătățirea relațiilor cu clienții – calculatorul poate fi mai ușor “educat”;
- Dezvoltarea de noi produse se poate realiza în concordanță cu cerințele consumatorului, monitorizând atitudinea acestora față de anumite produse/servicii;
- Noi modele de afaceri – bazele de date și preferințele clienților pot constitui informații importante pentru experții în finanțe.

Problemele sistemelor Comerț Electronic (E-Commerce)

- Asistența în unele cazuri poate fi deficitară iar informațiile furnizate despre un produs sau serviciu pot fi incomplete;
- Credibilitatea scăzută;
- Fiabilitatea pe termen lung este scăzută datorită dinamicității cu care se dezvoltă Comerțul Electronic (E-Commerce).

Scurt istoric (1)

- Prima tranzacție completă, realizată prin telefon/fax = **prima tranzacție de comerț electronic ???**
- 1960: (rețele private)
 - EDI (Electronic Data Interchange) – schimbul de documente standardizate
 - EFT (Electronic Funds Transfer) – fonduri dirijate electronic între organizații
- 1991: NSF – set de restricții privind utilizarea Internetului în scopuri comerciale
 - “dot-coms” și Internet “start-ups”
 - Vânzări directe online, e-Learning

Scurt istoric (2)

- 1992 - Compuserve oferă clienților săi posibilitatea de a cumpăra produse online cu ajutorul calculatoarelor proprii.
- 1994 - momentul nașterii Netscape-ului: browser simplu și Secure Sockets Layer (SSL).
1995 – apar Amazon.com și eBay.com.
- 1999 - valoarea cheltuită pentru achizițiile cu amănuntul pe Internet a atins cifra de 20 miliarde USD.
- 2000 - AOL și Time Warner fuzionează - o combinație a unui titan al vechii economii cu unul din noua economie digitală.
- 2001 - trecerea de B2B la B2E (business-to-employees) → mCommerce, eLearning.
2001 – rată accentuată de încetare a activității dot-com-urilor.
- Astăzi:
 -
 - - comerțul electronic - experiment consolidat;
 - cele mai pure companii de comerț electronic au devenit operațuni extensibile și generează creșterea vânzărilor (ex: Amazon.com).

Tipuri de site-uri Comerț Electronic (E-Commerce) (1)

- Stadiul 1 – companii orientate spre produs
 - broșuri: folosesc o cale de comunicare cu un singur sens
- Stadiul 2 – companii orientate spre vânzări
 - informații privind produsele și serviciile oferite, inclusiv modul și locul unde un potențial cumpărător poate să le achiziționeze
- Stadiul 3 – companiile care se bazează pe date
 - operațiuni comerciale pe site-uri, dar fără conectarea lor online cu sistemele offline.
 - preiau comenzi online, au o secțiune de întrebări adresate frecvent și admit comunicațiile prin e-mail pentru solicitări de produse sau servicii.
 - cele mai frecvent întâlnite

Tipuri de site-uri Comerț Electronic (E-Commerce) (2)

- Stadiul 4
 - Bazele de date online și offline pot fi conectate între ele (site-ul unui producător de automobile)
 - Utilizatorii pot personaliza propriul profil
- Stadiul 5
 - Sisteme de personalizare a experienței pentru utilizatori (servicii de chat și operatori vocali online)
 - Crearea și întreținerea costă mult

Comerț electronic vs. tradițional

- Comerțul electronic se deosebește de cel tradițional prin:
 - transformarea spațiului comercial (intermediarii tradiționali sunt înlocuiți; apar piețe și produse noi; se dezvoltă noi relații mai directe și mai intense între întreprinderi și între acestea și consumatori; apar noi locuri de muncă; funcțiile și competențele salariaților sunt redefinite)
 - manifestarea ca un catalizator (difuzarea pe scară largă în organizații a schimbărilor informaticе, comunicаtionale, manageriale și economice)
 - dezvoltarea interactivității între organizații și la nivel macroeconomic (comunicațiile și tranzacțiile economice se vor putea efectua în orice loc și în orice moment, determinând eroziunea frontierelor geografice și economice dintre organizații și țări)
 - adoptarea generalizată a Internetului asigură baza pentru tranzacțiile comerciale electronice (datorită utilizării de norme care nu sunt exclusive și a caracterului său deschis)

Comerț electronic și parțial

- Comerțul electronic pur vs. comerțul electronic parțial depinde de nivelul de digitizare a trei factori:
 - Produse (servicii) vândute.
 - Procesul de vânzare.
 - Agentul de vânzare (sau intermediarul digital).

Sistemul informatic

- O afacere de comerț electronic se sprijină pe un sistem informatic centrat Web, construit în jurul site-ului care conține magazinul on-line.
- În cadrul relațiilor comerciale clasice, tranzacțiile și achizițiile majore implică următoarele faze:
 - faza precontractuală, când se analizează oferta de pe piață;
 - faza contractuală, când se negociază clauzele contractuale (condiții de achiziție, servicii comerciale oferite, etc.);
 - faza realizării comenzi, când se identifică marfa, se verifică solvabilitatea clientului, etc.;
 - faza logistică, de pregătire și expediere a mărfii;
 - faza de achitare, care depinde de modalitatea de plată pentru care s-a convenit;
 - faza post-proces, care include servicii comerciale post-vânzare.

Fazele procesului de cumpărare

- Faza logistică nu are corespondent diferit în sistemul de comerț online
 - furnizarea bunului achiziționat este singura operație care se realizează prin metode tradiționale, fără a folosi instrumente specifice mediului Internet.

Comerțul tradițional

Faza precontractuală

Faza contractuală

Faza realizării comenzi

Faza logistică

Faza de achitare

Faza post-proces

Comerțul electronic

Gestionarea catalogului de produse

Gestionarea coșului de produse

Prelucrarea comenzi

Faza logistică

Sistemul de plăți electronice

Urmărirea relațiilor cu clienții

Câteva modele de afaceri on-line

- Magazin electronic (e-shop);
- Magazin electronic universal (e-mall) – o colecție de magazine virtuale reunite într-un site;
- Aprovizionarea electronică (e-procurement) – ex.: e-licitații.ro
- Comunități virtuale (virtual communities) – accesul se face pe bază de plată și se pot vinde/cumpăra produse doar între membri;
- Brokerajul de informații: cataloage de clienți clasificați pe profil, vânzarea de oportunități de afaceri, consultanță, etc.

Procesele implicate în dezvoltarea unui sistem de comerț electronic

Dezvoltarea unui sistem de comerț electronic presupune un efort din partea unei echipe mixte:

profesioniștii economici ce aparțin organizației (care cunosc foarte bine organizația, scopurile acestuia și pot lua decizii)

profesioniștii IT.

Procese tehnice

Identificarea problemei

Analiza cerințelor

Proiectare

Construire

Testare

Implementare

Utilizare

Asigurare suport

Înlocuire/dezvoltare

Procese manageriale

Planificare

Control și luarea decizilor

Alocarea resurselor și configurare

Managementul riscurilor

Fazele implementării unui sistem de comerț electronic

- Proiectarea proiectului
 - Model (vânzări)
 - Planificare
- Dezvoltarea software
- Dezvoltarea conținutului
- Lansare (Deployment)
 - Găzduire (Hosting) și operațiile suportate
- Serviciile de tranzacții

Opțiuni de implementare

- In-house vs. Outsource

	In-house	Outsource
Project design	Marketing and IT staff	Business consulting
Software development	Custom development	Off-the-shelf software/web services
Content development	Marketing staff	Web developer
Hosting and operations	IT staff	Internet service provider/ASP
Transaction services	IT staff	Commerce service provider

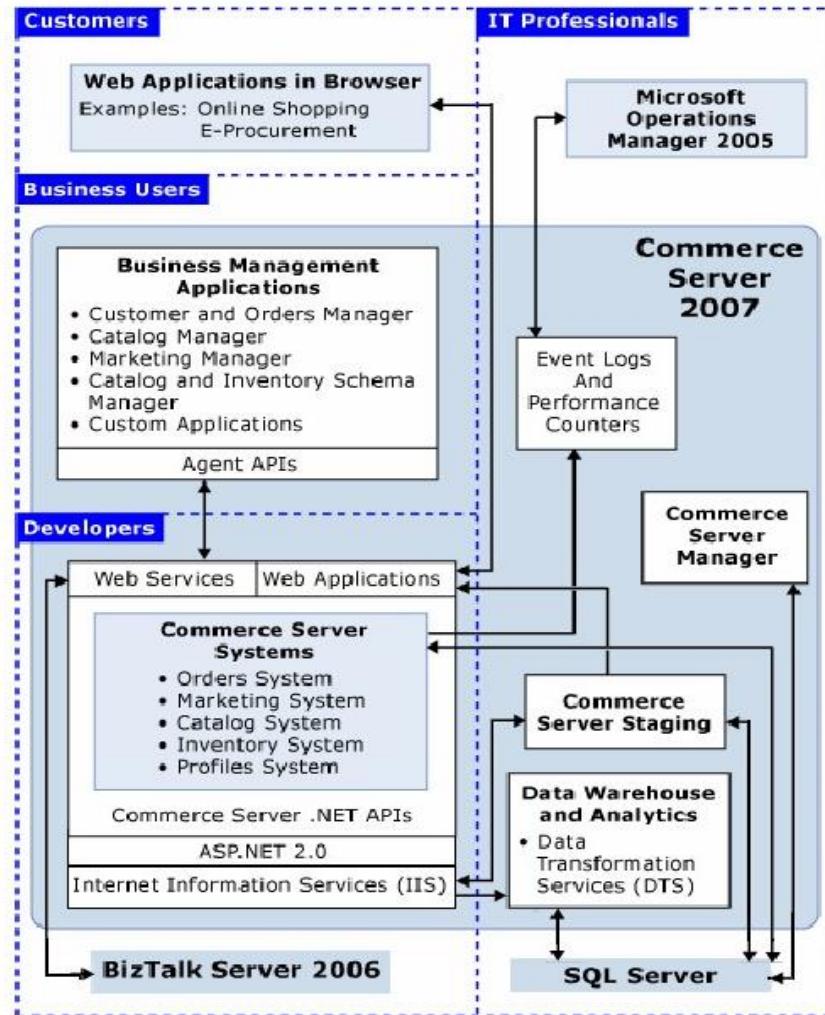
Comerț Electronic (E-Commerce) în România

 <p>www.okazii.ro licitatii online</p>	 <p>www.shopmania.ro comparatii intre diverse oferte ale magazinelor online</p>
 <p>www.emag.ro computere, gsm, electronice si accesorii</p>	 <p>www.culinar.ro super-market online pentru produse alimentare si de igiena</p>
 <p>www.price.ro comparatii intre produsele oferite de diverse magazine online</p>	 <p>www.dc-shop.ro magazinul online al firmei Depozitul de Calculatoare</p>
 <p>www.librarie.net librerie si anticariat online</p>	 <p>www.domo.ro magazin online de produse electrocasnice, electronice si IT&C</p>
 <p>www.dol.ro magazin de media (carte, muzica, film), IT si papetarie</p>	 <p>www.ultrapro.ro magazinul online al firmei UltraPro</p>

Studiu de caz: Microsoft Commerce Server 2007

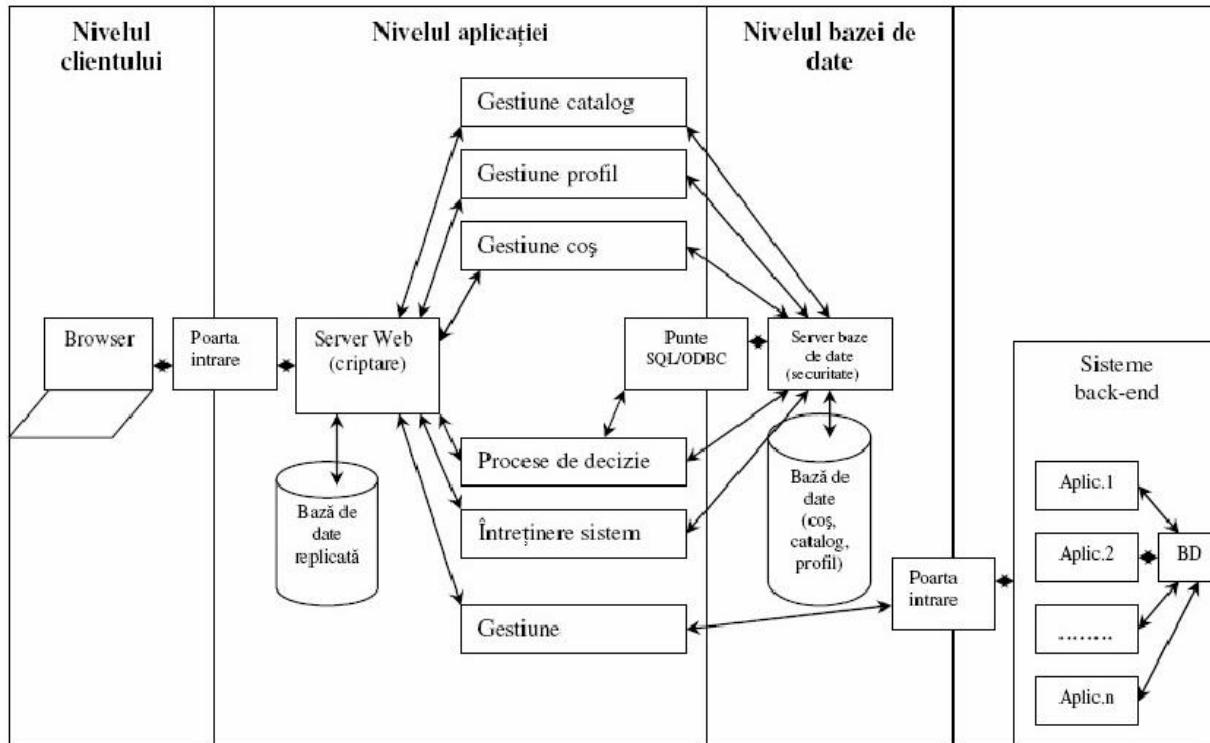
- Commerce Server 2007 este o soluție cuprinzătoare pentru multe tipuri de scenarii de business incluzând:
 - Vânzări B2C (Business to Consumer) pentru diferite bunuri sau servicii online
 - Scenarii B2B (Business to Business) între diverse comunități comerciale
 - Scenarii B2X care combină B2C și B2B
 - Servicii pentru portaluri care cuprind cataloage, profile, informații de targeting pentru livrările personalizate.

Arhitectura Commerce Server 2007



Concepțe arhitecturale ale sistemelor de E-Commerce

Aplicația de comerț electronic



Aplicația de comerț electronic

- Serverul Web
 - asigură funcționalitatea întregului pachet software al aplicației

- Gestionarea catalogului de produse
 - folosește datele din baza de date de produse pentru a genera catalogul cu oferta în format HTML
 - utilizatorii pot consulta catalogul prin intermediul browser-ului pentru a căuta și compara produsele din oferta

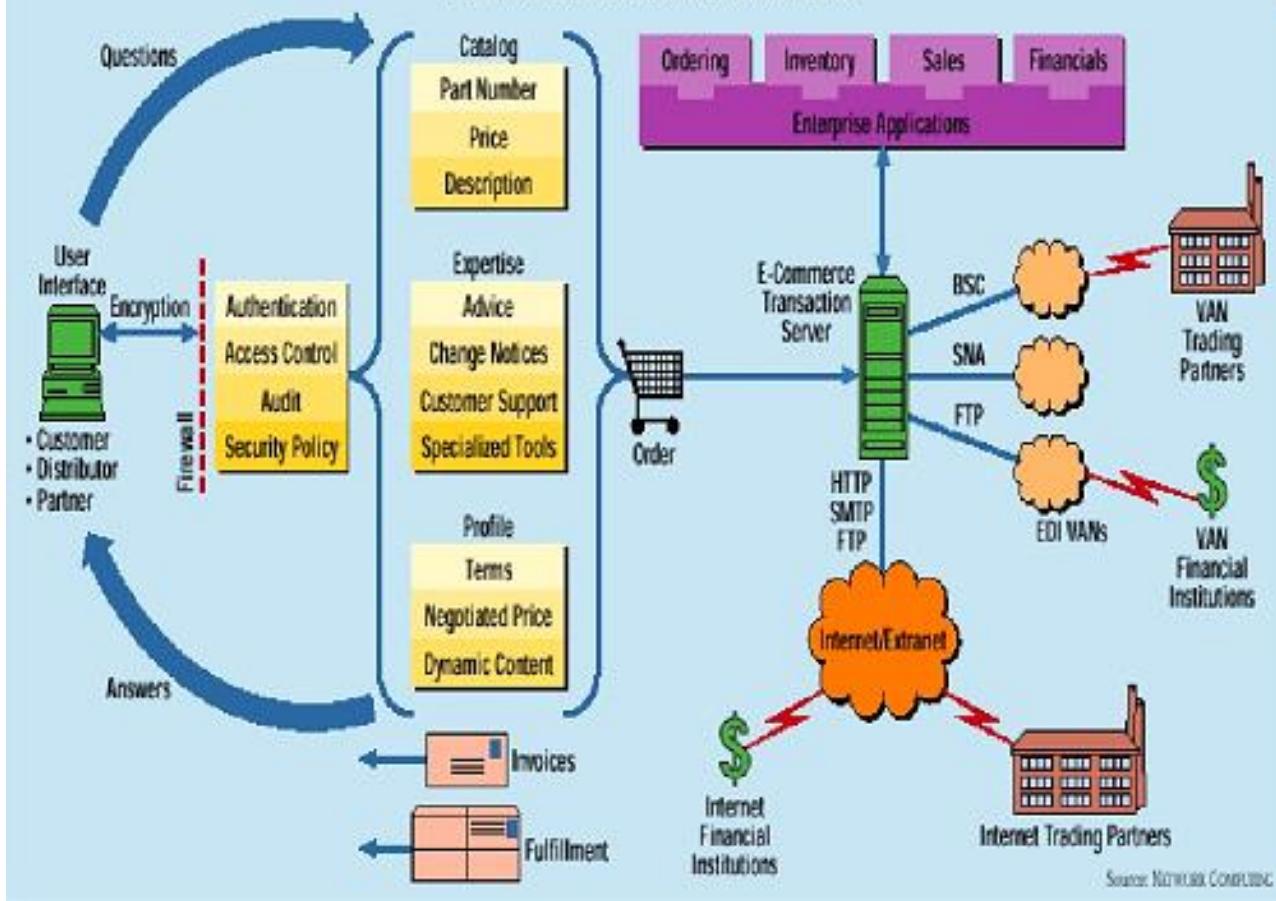
- Coșul de cumpărături virtual
 - modalitatea standard de gestionare a produselor selectate de client și de lansare a comenzi

- Baza de date de produse
 - localizată pe serverul corespunzător
 - este posibil ca aceleași date să fie replicate în baze de date pentru utilizator, mai accesibile aplicației care gestionează magazinul online

Aplicația de comerț electronic

- **Componenta de prelucrare a comenzi** (ordinul de cumpărare)
 - are implementate regulile de derulare ale afacerii (tranzacția comercială):
 - consultarea și actualizarea bazei de date de produse
 - calculul prețului
 - selectarea modalității de livrare
 - a modalității de plată
 - a serviciilor de securitate folosite
 - Aplicația interfațează cu aplicațiile sistemului tradițional de urmărire a activității comerciale:
 - gestiunea stocurilor
 - urmărirea comenzilor
 - calculul prețurilor
 - urmărirea încasărilor
 - relațiile cu terții, etc.
- **Componenta pentru urmărirea relațiilor cu clienții**
 - gestionează profilul utilizatorilor, pe baza datelor transmise de vizitatorii site-ului și/sau pe baza datelor privind comportamentul de vizitare a site-ului, preluate, de obicei, din jurnalele de activitate gestionate de serverul Web
- **Componenta pentru întreținerea sistemului**
 - trebuie să ofere persoanelor autorizate posibilitatea de dezvoltare sau reconfigurare a sistemului, într-o manieră cât mai accesibilă și mai sigură, de preferat prin intermediul browser-ului

E-Commerce From The Seller's Point Of View



Source: Network Computing

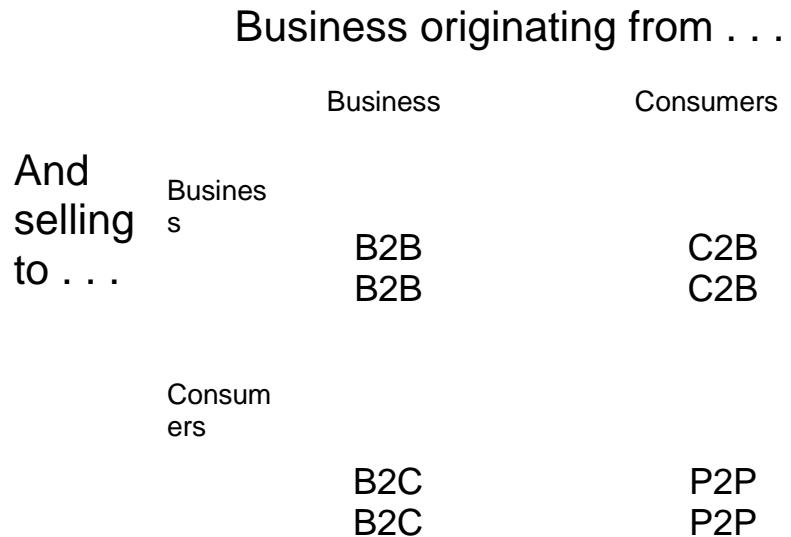
Modele e-business

- 2 perspective: tranzacția de cumpărare și cea de vânzare
 - 2 tipuri majore de comerț electronic
- “business-to-consumer (B2C)”
 - Tranzacțiile online sunt realizate între comercianți și consumatori individuali
- “business-to-business (B2B)”
 - Afacerile economice sunt realizate prin tranzacții online cu alte firme
- Mai nou “Business-to-Employees (B2E)”:
 - Model de comerț electronic în care o organizație livrează servicii, informații sau produse proprietilor salariați

Mecanisme de comerț electronic (1)

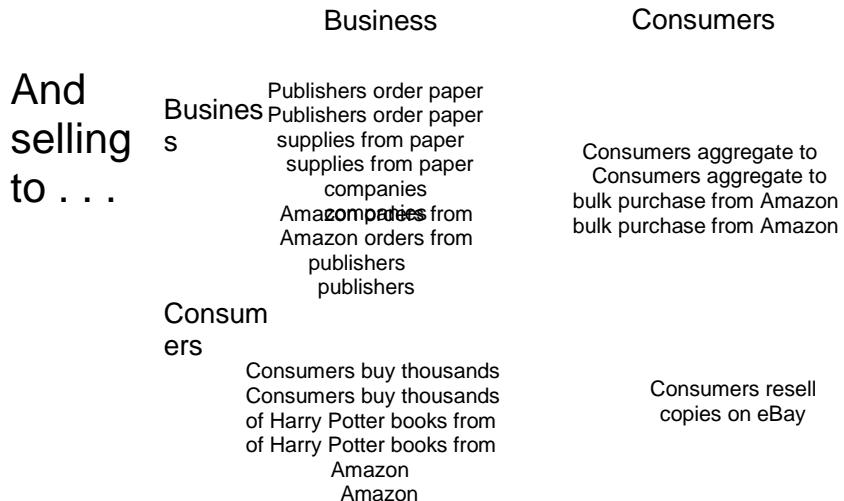
- “Business-to-Business-to-Consumer (B2B2C)”:
 - Model de comerț electronic în care un partener de afaceri furnizează câteva produse sau servicii pentru un partener client care menține proprii clienți
- “Consumer-to-Business (C2B)”:
 - Model de comerț electronic în care indivizi utilizează Internet-ului pentru a vinde produse sau servicii organizațiilor sau indivizi care caută vânzători pentru a ofera produse sau servicii de care ei au nevoie.
- “Consumer-to-Consumer (C2C)”:
 - Model de comerț electronic în care consumatorii vând direct alțor consumatori (licitații online, comunități virtuale).
- “Mobile commerce (m-commerce)":
 - Tranzacții de comerț electronic și activități conduse într-un mediu wireless.
- “Location-based commerce (l-commerce)":
 - Tranzacții tip m-commerce având drept țintă indivizi în locații specifice, la un timp specificat.
- “Collaborative commerce (c-commerce)":
 - Model de comerț electronic în care indivizi sau grupuri de indivizi sau grupuri de comunicare efectuează colaborări online.
- “e-exchange (bursă electronică)":
 - O piață electronică publică cu mulți cumpărători și vânzători.
- “Exchange-to-exchange (E2E)":
 - Model de comerț electronic în care schimburile electronice formale se conectează unele cu altele în scopul schimbului de informații.

Mecanisme de comerç electronic (2)



Mecanisme de comerç electronic (3)

Business originating from . . .



Convergența categoriilor sistemelor e-commerce

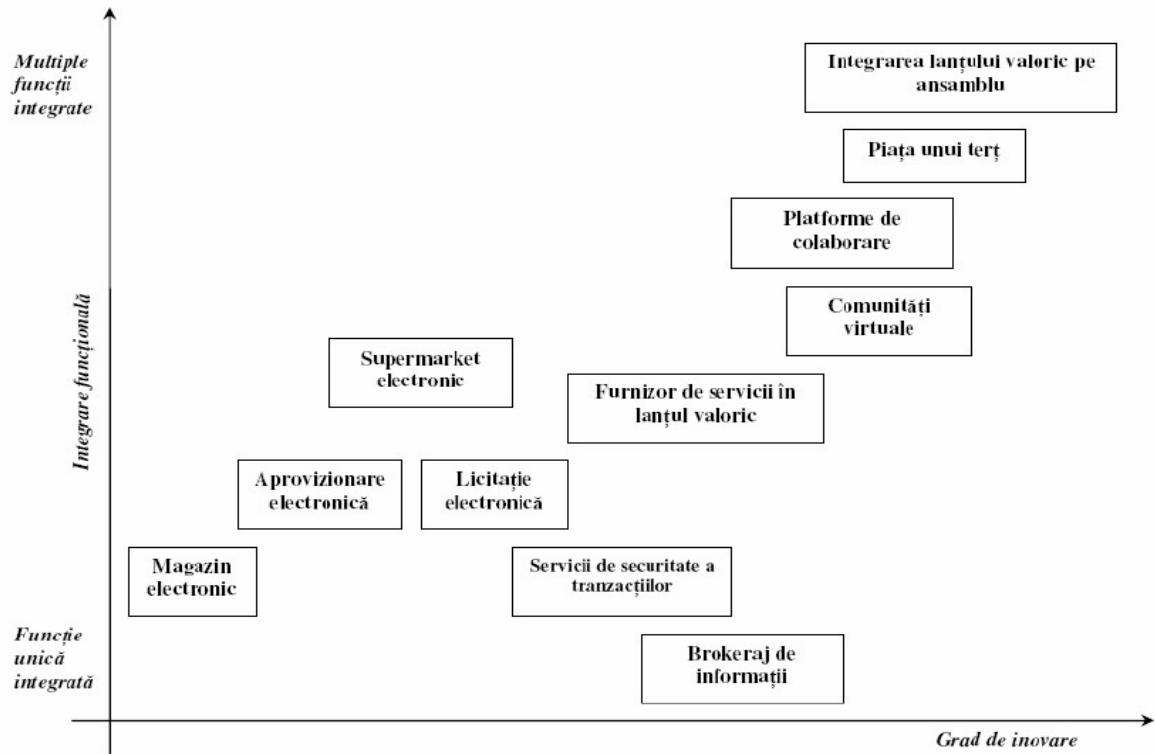
Modele de afaceri

- Comerțul electronic - modalitate complementară alături de cel tradițional
- Ce tipuri noi de modele de afaceri pot apărea?
- Ce strategie de marketing se poate aplica în aceste cazuri?
- **model de afaceri** - o arhitectură pentru produse, servicii, fluxuri de informații, inclusiv o descriere a participanților la procesele de afaceri, rolurile acestora; o descriere a potențialelor beneficii cât și a surselor de venituri

Modele de afaceri

- Pentru evaluarea unui model de afaceri specific comerțului electronic trebuie să se aibă în vedere următoarele considerații de piață:
 - mecanismul pe care îl adoptă firma: B2B, B2C, etc.,
 - poziționarea firmei pe lanțul valoric între clienți și furnizori,
 - declarația de valoare și grupul țintă de clienți căreia i se adresează,
 - modelul specific de generare a fluxului de venituri,
 - reprezentarea firmei în lumea fizică și virtuală.

Modele de comerț electronic



Modele de afaceri (1)

- **e-shop** (magazinul electronic):
 - promovează produsele/serviciile firmei,
 - beneficiile firmei: creșterea cererii și prețuri promotional scăzute,
 - beneficiile clienților: disponibilitate 24 de ore, prețuri mai mici decât cele tradiționale, informare mai bună, service îmbunătățit, veniturile sunt obținute din vânzări mari, costuri scăzute, promovare.
- **e-procurement** (aprovisionare electronică):
 - permite realizarea achizițiilor electronice de produse/servicii,
 - multiplele posibilități de alegere a furnizorilor, ceea ce duce la costuri scăzute de achiziție, calitate îmbunătățită și îmbunătățirea distribuirii
veniturile provenind din reducerea costurilor.
 -
- **electronic auctions** (licitații electronice):
 - implementarea electronică a mecanismului licitațiilor tradiționale,
 - veniturile provin din taxe pe valoarea tranzacțiilor și publicitate.
- **e-mall** (mall electronic):
 - un set de e-shop-uri.
- **third party marketplace** (piata unui terț):
 - îndeplinește funcția de promovare/vânzare asigurate de terți,
 - promovarea numelui, asigurarea logisticii pentru efectuarea plăților sau prelucrarea comenziilor, securitatea tranzacțiilor,
veniturile provin din taxe de membru, taxe de servicii, taxe pe tranzacții.
 -

Modele de afaceri (2)

- **virtual community** (comunități virtuale):
 - construiesc profilul clienților,
 - beneficiile: creșterea atractivității serviciilor și promovare, veniturile provin din taxe de membru.
- **value chain service provider** (furnizor de servicii în lanțul valoric):
 - îndeplinesc o funcție specifică din lanțul de distribuție (asigurarea logisticii, a platilor electronice)
 - veniturile provin din taxe de servicii.
- **value chain integrators** (integrarea lanțului valoric):
 - focalizate pe integrarea mai multor pași din lanțul de distribuție,
 - veniturile provin din taxe de consultanță sau taxe de tranzacții.
- **collaboration platform** (platformă de colaborare):
 - asigură instrumente și mediu informațional pentru colaborarea între firme,
 - veniturile provin din taxe de membru, de servicii oferite, prețul utilizării platformei.
- **brokeraj de informații, servicii de securitatea tranzacțiilor** (trust services, infobrokerage):
 - furnizează informații și consultanță în diferite domenii, oferă siguranță tranzacțiilor
 - veniturile provin din taxe de subscrisoare, vânzări de software.

Concepte arhitecturale

- Arhitectura unui sistem: „organizarea fundamentală a unui sistem, încorporată în componentele și relațiile dintre acestea și în principiile care ghidează conceperea și evoluția sa”
 - Componente de bază, concepte, relații, constrângeri
- Arhitectura “consumer retail” ≠ “publishing system” (procese de afaceri diferite)
- Mai multe asemănări decât deosebiri între sistemele de comerț electronic → nucleu de arhitectură (core architecture)
 - reutilizarea arhitecturii și a muncii de proiectare dezvoltate anterior

Arhitectura unui sistem (1)

- Caracteristici:
 - funcționale – setul complet de funcții ale sistemului cerute de utilizatori
 - non-funcționale – caracteristicile de securitate, disponibilitate, performanță → constrângeri impuse și atribute de calitate
 - de implementare – evolutivitate, reutilizabilitate, costuri și integritate

Arhitectura unui sistem (2)

- Definirea unei arhitecturi urmărește:
 - **Complexitatea**: maniera de descompunere și modularizare a sistemului
 - **Evolutivitatea**: schimbările survenite în activitatea firmei utilizatoare, fie de natură strategică, organizațională sau procedurală
 - **Scalabilitatea**: abilitatea sistemelor distribuite de a permite adăugarea de noi posturi client sau server, fără o modificare semnificativă a performanțelor de ansamblu ale sistemului

Arhitectura unui sistem (3)

- **Arhitectura logică:** modul în care se structurează și se organizează sistemul la nivel conceptual pentru a asigura funcționalitățile cerute de utilizator.
- Arhitectura pe 3 nivele:
 - prezentarea (interface)
 - logica problemei sau a activității de afaceri (business)
 - gestiunea datelor (persistence)
- **Arhitectura fizică:** modul în care se implementează fizic structurile definite în cadrul arhitecturii logice (structura programelor și distribuirea lor pe echipamentele utilizate).

Arhitectura client-server

- Distribuirea funcționalităților sistemului între client și server.
- Arhitectura tradițională pe două straturi – two-tier client/server
- Serverul și clientul comunică direct (highly coupled manner)
- Thin-client
 - Clientul implementează doar stratul de interfață a aplicației software
 - Avantaje
 - concentrarea logicii proceselor de afaceri doar pe server
 - utilizarea de echipamente hardware modeste pentru mașina client
 - Dezavantajul major este dependența crescută de serverul unde sunt concentrate procesele de bază ale aplicației software.
- Fat-client
 - Mașina client implementează atât stratul de interfață, cât și logica proceselor de afaceri ale aplicației software
 - Avantaj - transfer a unui număr mare de prelucrări pe mașina client, fructificând puterea de procesare a acestuia.
 - Dezavantajele:
 - necesitatea unor echipamente hardware puternice pentru client
 - întreținere/administrare/actualizare dificilă a codului de program pentru client
 - intensificarea traficului de date prin rețea datorită procesării reduse a acestora pe server.

Arhitectura client-server

- Arhitectura pe trei straturi – three-tier client/server
 - Server de aplicație - interfață către alte servere din rețea
 - Logica afacerii din cadrul aplicației se află într-un strat propriu
 - Avantaje:
 - Flexibilitate
 - Reducerea timpului total de răspuns unei cereri client
 - Reducerea traficului de date prin rețea
 - Îmbunătățirea eficienței aplicațiilor software
 - Dezavantaj - serverul de aplicație
- Arhitectura client/server multi-nivel – n-tiers client/server
 - Mai multe servere distribuite în rețea, atât pentru logica afacerii, cât și pentru date

Arhitectura Web

- Client-server peste web:
 - interactivitate,
 - scalabilitate,
 - asigurarea de pârghii de comunicare cu sisteme existente,
 - costuri de administrare zero
- Integrarea sistemelor pe 3 straturi cu (cel puțin) tehnologiile Web
 - Browser web = client thin
 - Server web = server de aplicații
- Evoluție arhitecturală la medii distribuite n-straturi
 - Stratul din mijloc poate fi divizat în server web și server de aplicații

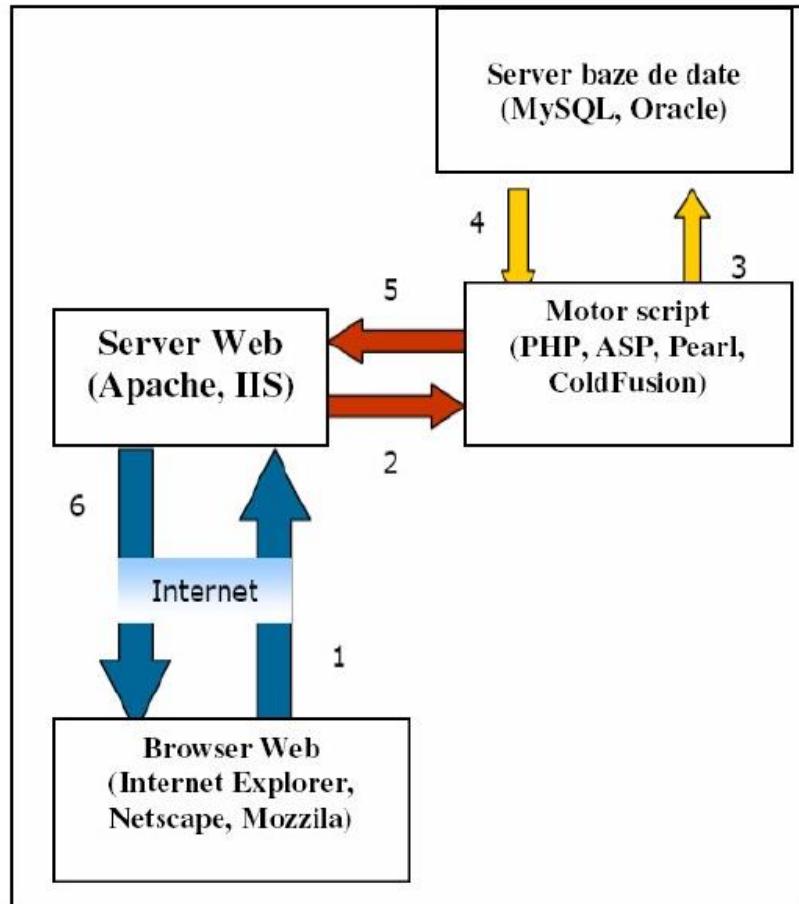
Arhitectura multistrat

- Stratul client
- Stratul Web
- Stratul business
- Stratul sistemului informațional intern

Arhitectura I-net (1)

- Tip hibrid de arhitectură (client/server și host)
- Sistem client/server, care folosește browserul Web drept client și un server Web drept componentă back-end
- Flux de date bidirecțional între client și serverul de aplicație (clientul poate modifica conținutul Web)

Etapele unei tranzacții Web



Arhitectura I-net (2)

- Avantaje:
 - Costuri reduse
 - Aplicații pentru mai multe platforme
 - Configurare simplă
 - Tehnologii bazate pe standarde deschise
 - Ușurință în utilizare
 - Control centralizat al informației
 - Administrarea simplă a resurselor

Arhitectura generală a unui sistem de comerț electronic

- Un sistem de comerț electronic conține:
 - Stratul de prezentare (Presentation layer)
 - Nivelul afacerii (Business tier)
 - Nivelul bazei de date (Database tier)
- Reutilizarea soluțiilor anterior validate =>
șablon

Arhitectura funcțională

- Etapele proiectării arhitecturii unui sistem:
 - Identificarea rolurilor
 - Descompunerea în componente
- În plus, pentru sistemele de comerț electronic:
 - Procesarea tranzacțiilor electronice
- Întrebări frecvente:
 - Cum se declanșează o tranzacție de către utilizatori?
 - Cum se verifică informațiile transmise de utilizatori?
 - Cum se potrivesc cererea cu oferta?
- Specificarea tipurilor de relații între componentele sistemului => trust model

Sumar

- Modele de comerț electronic
- Arhitecturi de sisteme de comerț electronic

Arhitecturi E-Commerce

Arhitectura funcțională

- Etapele proiectării arhitecturii unui sistem:
 - Identificarea rolurilor
 - Descompunerea în componente
- În plus, pentru sistemele de comerț electronic:
 - Procesarea tranzacțiilor electronice
- Întrebări frecvente:
 - Cum se declanșează o tranzacție de către utilizatori?
 - Cum se verifică informațiile transmise de utilizatori?
 - Cum se potrivesc cererea cu oferta?
- Specificarea tipurilor de relații între componentele sistemului => trust model

Identificarea rolurilor

- **Client** (customer role)

- Specificator, aprobator, cumpărător, recipient
- Anonymous buyer, member customers (member administrator)

- **Vânzător** (business role)

- Echipa de conducere (business manager, arhitectul sistemului, proiectantul de conținut, autorul de conținut, implementatorul, administratorul bazei de date, echipa de vânzări și marketing)
- Echipa operativă (managerul operativ, supervisorul de sistem, administratorul de sistem, ofițerul de securitate, agentul de livrare, contabil)

Descompunerea funcțională în componente

- Componența **cumpărător**
 - Uzual browser web
 - Aplicații client particulare (client wallets, server-side wallets, buyer home community)
- Componența **vânzător**
 - Content management system
 - Transaction processing system
 - Payment processor
 - Fulfillment systems
- Grupare logică:
 - Front office (marketing și vânzări)
 - Back office (gestionarea tranzacțiilor)

Rolul arhitecturilor bazate pe componente

- Sistemele de comerț electronic sunt utilizate de aproape trei decenii
- Dificultăți în adaptarea la schimbările rapide ce au loc în mediul de afaceri
- Sistemele tradiționale tind să fie dependente de o anumită platformă de operare

Etapele arhitecturilor de comerț electronic

Analiza arhitecturilor de comerț electronic

- Patru componente:
 - Clientul
 - Vânzătorul
 - Sistemul tranzacțional
 - Portal de plată (Payment gateway)

Merchant server

- Combină componentele vânzător și sistemul tranzacțional
- Avantaj: simplitate
- Dezavantaj: extindere dificilă, imposibilitatea utilizării de noi tehnologii

A

Structura logică

Arhitectura comerțului distribuit în piață deschisă (Open Market Distributed Commerce Architecture) (1)

- Separă managementul de conținut de cel al tranzacțiilor – **SecureLink**
- Secure Remote Procedure Call bazat pe HTTP și HTML
- Interoperabilitate
- MAC (message authentication code)
- Componente furnizate
- Gestiunea cheilor
- Kit de dezvoltare software
- Ex:
`http://payment.tscaorp.com:80/bin/bayment.cgi?ac1c7b48
9d400e4a98a6e9c8b9851a37:kid=196003.190007&valid=
815499241&expire=....`

<http://www.securepay.com.au/securepay/products/online/securelink.shtml>

Arhitectura comerțului distribuit în piață deschisă (Open Market Distributed Commerce Architecture) (2)

- Serverul de tranzacții este separat de cel al vânzărilor
- Payment gateway poate fi separată sau nu în funcție de metodele de plată online
- Avantaje:
 - permite catalogage multiple pe mai multe servere
 - separarea componentelor client de cele de tranzacții
 - gestionarea separată a serviciilor de securitate

Architectura OBI (Open Buying on the Internet)

- Standard propus de OBI (Open Buying on the Internet)
- Standard în domeniul aplicațiilor B2B
- Separarea funcționalității sistemului:
 - Activități de cumpărare
 - Activități de vânzare

Structura logică

- Cumpărare:
 - Baza de date clienți
 - Profilurile clientilor
 - Procesele de aprobare
- Vânzare:
 - Catalogul
 - Gestionarea ordinelor
 - Activități de plată
 - Realizare servicii pe partea de vânzare

Fluxul tranzacțiilor - Open Buying on the Internet

Arhitectura OBI

Arhitectura tehnică (1)

- Arhitectura aplicației
 - Server web
 - Pagini web
 - Componente middleware
 - Obiecte business
 - Baze de date
- Infrastructura acesteia
 - Echipamente hardware
 - Elemente de rețea
 - Firewalls
 - Routere

Arhitectura tehnică (2)

Componentele arhitecturii tehnice

- Cerințe funcționale:
 - Disponibilitate
 - Performanță
 - Securitate
 - Integritatea tranzacțiilor
 - Scalabilitate
 - Robusteză
 - Interoperabilitate
- Performanța end-to-end: viteza reală de răspuns obținută la cererea unui singur utilizator
- Scalabilitatea: sistem funcțional și disponibil (QoS) și în cazul creșterii numărului de utilizatori

Dezvoltarea bazată pe componente (economice)

- O componentă economică = o implementare software a unui concept economic autonom sau a unui proces economic.
- Consta din toate elementele software necesare pentru a reprezenta, implementa și desfășura un concept într-un mod autonom, astfel încât să fie reutilizabil pentru alte produse software.

Perspectiva dezvoltării bazată pe componente

- Planificarea resurselor întreprinderii (Enterprise Resource Planning - ERP)
 - Integrează principalele procese economice (finanțe, managementul resurselor umane, vânzări, etc.)
- Managementul relațiilor cu clienții (Customer Relationship Management - CRM)
 - Cuprinde toate interacțiunile dintre o companie și clienții/partenerii săi (prin Internet, telefon, ATM/Kiosk)
- Comerț electronic
 - Permite accesul pe web al sistemelor ERP
- Business Intelligence (BI)
 - Aplicații pentru colectarea, stocarea, analiza și descoperirea anumitor aspecte care ajută luarea unor decizii manageriale (Decision Support Systems, Online Analytical Processing, Data Warehouse, Data Mining)

Relațiile dintre componente

Avantajele unei abordări bazată pe componente

- Încapsularea implementării
- Utilizarea unei metodologii pentru crearea, proiectarea și dezvoltarea de aplicații pentru afaceri economice
- Costuri reduse
- Oferă reguli și structuri ce permit:
 - Flexibilitate
 - Posibilitatea de integrare a altor componente

Nivelele arhitecturii bazată pe componente

- Nivelul interfață utilizator (user interface layer)
- Nivelul economic (business layer)
 - Componente ale procesului (process components)
 - Componente ale domeniului economic (business domain components)
 - Componente ale infrastructurii afacerii (business infrastructure layer)
- Nivelul infrastructurii tehnice (technical infrastructure layer)

Concentrația eforturilor
de dezvoltare

Arhitectura unei aplicații

Pentru a beneficia de avantajele reutilizării, arhitectura unei aplicații ar trebui să cuprindă aceste nivele

Arhitectura unei aplicații

- Procese economice de bază
 - Set de blocuri constructive
- Funcționalități comune mai multor domenii economice
 - Obiecte economice (adresa, parteneri, metode de plata, etc.)
 - Obiecte ce manipulează obiecte economice (conturi bancare, parteneri, metode de plata)

Analiza și proiectarea unui sistem de comerț electronic

Analiza și proiectarea sistemelor de comerț electronic (1)

- Productivitatea și calitatea impune o atenție deosebită acordată activităților de analiză și proiectare
- Diverse metode de analiză și proiectare
- Diverse școli (metode de analiză și proiectare):
 - “scoala descompunerii funcționale” (modularizarea programelor – fiecare modul responsabil de o funcție a sistemului)
 - Baza modulelor sunt datele (un modul încapsulează o structură de date - aplicații real-time: un modul recunoaște și răspundă unui eveniment)

Analiza și proiectarea sistemelor de comerț electronic (2)

- În varianta orientată-obiect, orice componentă poate fi încapsulată ca un obiect reutilizabil
- Preocupări actuale în proiectare și analiză:
 - Utilizarea pe scară largă a tehnologiei CASE
 - Perfecționarea și standardizarea metodologiei orientată-obiect
 - Introducerea abordărilor soft
 - Utilizarea ontologiei

Conceptul CASE

CASE = Computer Aided Software Engineering =
Ingineria Proramării Asistată de Calculator

Software
Engineering

Computer aided

Foundation (Formalization)

Methods

Tools

Complex methods become
easy to use

CASE

Abordări soft

- Concept dezvoltat de Peter Checkland
- Scopul îl reprezintă reprezentarea perspectivelor multiple pe care utilizatorii le au asupra sistemului

Sisteme soft vs. hard

- E dificil de produs un sistem bine-definit (“hard”) pentru a
- În multe situații reale lucrurile nu sunt bine definite:
 - “Vezi ce poți face pentru a implementa o soluție cât mai eficientă”
 - “Vino cu un plan și prezintă-l”
 - “Vezi ce faci și îmbunătășește performanțele proiectului”
- Activitățile de proiectare “hard” se concentrează pe “cum”:
 - Cum poate fi făcută asta? (proiectare)
- Metodologiile de proiectare “soft” adresează problema:
 - “Ce este necesar?” (analiza necesităților)

Utilizarea ontologiilor

- Ontologia = specificație formală, explicită având ca scop o conceptualizare partajată
- Îmbogățirea paginilor web cu informații va permite urmărirea și interpretarea informațiilor =>
 - dezvoltare de servicii Web automatizate
 - agenti funcționali

Abordarea orientată obiect

- Organizarea și funcționarea programelor, aplicațiilor sau sistemelor informaticice ca un ansamblu de obiecte distincte ce cooperează între ele
- Avantaje:
 - Capacitatea de modelare a obiectelor complexe
 - Reducerea neconcordanțelor între lumea reală și sistem
 - Încapsularea acestor obiecte și a serviciilor realizate
 - Starea obiectelor poate fi modificată rapid
 - Sistemele au grad mare de adaptabilitate la schimbare
- Metode de analiză și proiectare orientată-obiect
=> standardizarea în limbajul UML

UML (Unified Modeling Language = Limbaj unificat de modelare)

- Limbajul unificat de modelare (adoptat în 1997 de OMG drept limbaj standard de modelare obiectuală)
- Permite analiza și proiectarea sistemelor informaticе folosind conceptele orientate-obiect
- Notația folosită în modelare este formată din simboluri grafice
- Definește multiple vederi asupra sistemului:
 - Diagramе statice (diagramе de situație și diagramа claselor)
 - Diagramе de comportament (diagramа de stare, de activități, de interacțiune)
 - Diagramе de implementare (diagramа componentelor, de configurație și a pachetelor)

Ierarhia de modele, vederi și diagrame utilizate în limbajul UML

UML asigură traiectul

modele → vederi → diagrame → fișiere de cod sursă → date/cazuri de test

Abordarea etapelor de analiză și proiectare

- Abordarea modernă a arhitecturii sistemelor complexe (Meyer, Rumbaugh, Jacobson, Booch, etc.): Concepția arhitecturii sistemului este o activitate primordială în raport cu etapa de implementare, privită ca activitate secundară
- Conceptul a dus la:
 - Mecanisme şabloane de proiectare
 - Diverse instrumente CASE

RUP

- Rational Unified Process – proces general pentru dezvoltarea orientată obiect de produse informaticе
 - Instrument CASE ce utilizează UML – Rational Rose
- Nucleul îl reprezintă metodologia Unified Process → permite dezvoltarea de metodologii pe diverse tipuri de sisteme informaticе
- Ultima versiune include și extensii/notații specifice dezvoltării aplicațiilor Web (User Interface Guidelines)

Metodologii de realizare a sistemelor de comerț electronic

- Modelare Web (perspectiva logică)
 - Stratul de prezentare
 - Stratul business
 - Stratul informațional intern
- Stratul de prezentare rulează pe partea client
- Stratul de business pe server
 - Stratul Web
 - Stratul regulilor de afaceri

Aplicații stateless (independete de stare)

- Într-o aplicație distribuită un obiect instanțiat poate fi accesat oricând pe perioada execuției
- Un serviciu Web este **stateless**
 - Serverul pierde toate informațiile despre client odată cu închiderea conexiunii curente
- Soluții: gestionarea conexiunilor prin fișiere cookies
- **Obiecte tip session**
- **Hidden fields**

Şabloane primare pentru arhitecturi Web

- Thin client
 - Avantaj: viteza și portabilitatea
 - Dezavantaj: interfață
- Thick client
 - Avantaj: Interfață mai sofisticată, robustă și prietenoasă
 - Dezavantaj: viteza și portabilitatea

Procesul de proiectare

- Modelarea:
 - Definirea cazurilor de utilizare (use-case)
 - Definirea/documentarea cerințelor funcționale și non-funcționale
 - Descrierea fluxurilor de evenimente pentru cazurile de utilizare
- Proiectarea
 - Diagrame de colaborare și înlănuire ale obiectelor ce participă la un scenariu use-case
 - Problemă: **o aplicație Web este stateless...** Obiectele de pe un strat pot accesa obiectele de pe alt strat după închiderea conexiunii clientului?

Proiectarea

- Proiectanții pot folosi fișierele cookies sau obiecte tip session
- Fișierele cookies
 - conțin text ce poate fi regăsit ulterior
 - parte a headerelor HTTP

```
•<?php  
    setcookie("message", "No worries.");  
?>  
<html>  
    ...
```

- Obiecte session
 - rețin date pe partea de server a aplicației

```
session_start();  
$_SESSION['color'] = "blue";
```

Concepțe teoretice

- Structurarea sistemelor informaticice specifice comerțului electronic

Sistem informatic centrat Web

- Tipuri speciale de sisteme ce utilizează tehnologiile Web
- Dpdv al tehnologiilor Web utilizate:
 - Site-uri statice
 - Aplicații Web
- O aplicație Web
 - permite accesarea datelor complexe
 - oferă servicii interactive via Web ce pot modifica starea afacerilor

Sisteme informatiche de administrare

- Content management system (sistem de administrare a conținutului)
 - permite crearea, arhivarea, căutarea și controlul/publicarea informațiilor
- Web content management system (WCM)
 - automatizarea activităților: crearea și dezvoltarea de conținut
- Sisteme CMS centrate Web
 - sisteme particulare ale sistemelor WCM
 - managementul și controlul informațiilor dinamice ale unei societăți

Concepțe utilizate

- Metodologiile moderne de dezvoltare lucrează cu:
 - Procese
 - Metode
 - Tehnici
 - Activități
 - Proceduri
 - Notații
 - Modele
 - Diagrame

Metodologia de situație (1)

- Axată pe metodologia de realizare definită de cerințele proiectului dezvoltat
- Model cadru:
 - Strategia de asamblare bazată pe reutilizarea componentelor
 - Strategia de extindere
 - Strategia de concepere a noii metodologii prin abstractizarea modelelor obținute sau prin instanțierea unui meta-model

Metodologia de situație (2)

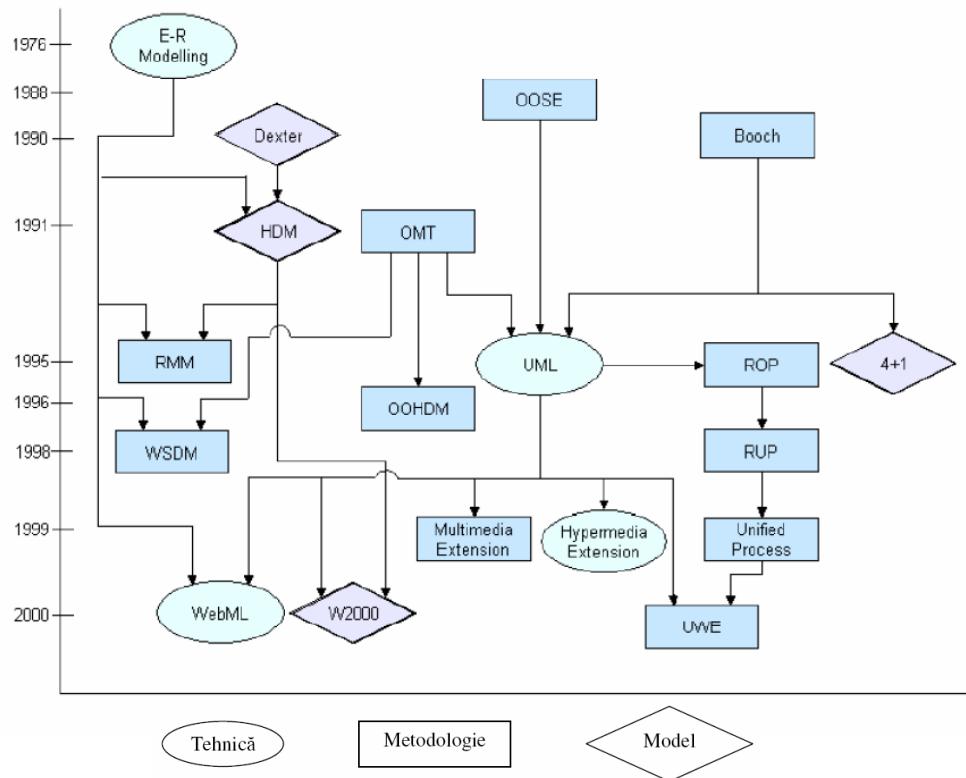
- Etape importante ale procesului:
 - Specificarea cerințelor impuse metodologiei de realizare
 - Selectarea părților de metodologii existente
 - Asamblarea acestora în vederea obținerii noii metodologii
- Metodologia de configurare: adaptarea unei metodologii existente la o nouă situație
- Metodologia de situație: se concepe pe baza unui set de metodologii existente și este specifică unei anumite situații

Metodologii de realizare a sistemelor informaticice

Introducere

- De-a lungul timpului au fost dezvoltate un număr impresionant de metodologii de realizare a sistemelor informatic Web
- Aplicațiile Web
 - Facilitează accesarea datelor
 - Oferă servicii interactive via Web
- Probleme legate de
 - Consistență a datelor
 - Navigare
 - Trasabilitate și mapare a fluxurilor de date Web pe procesele de afaceri dezvoltate
- Solutia: abordarea conținutului managerial al unui sistem informatic (CMS)
 - mediu integrat și flexibil de management și control al informațiilor

Evoluție istorică (1)



Evoluție istorică (2)

- (1976)
 - E-R Modeling – tehnica entitate-asociere
- (1988-1990)
 - Dexter – tehnica de modelare structurata propusa de Dexter
 - OOSE – Object-Oriented SoftwareEngineering
 - Booch – metodologie elaborata de Grady Booch similara metodologiei OMT
- (1991)
 - OMT – Object Modeling Technique
 - HDM – Hypermedia Design Method
- (1995)
 - UML – proces iterativ de dezvoltare a sistemelor informaticice utiliyând limbajul UML
 - 4+1 – modelarea prin cele 5 vederi (vederea logica, vederea distribuirii, vederea implementarii, vederea exploatarii, vederea cazurilor de utilizare)
 - RMM – Relationship Management Methodology, metodologie pentru realizarea aplicatiilor hypermedia
 - ROP – Rational Object Process

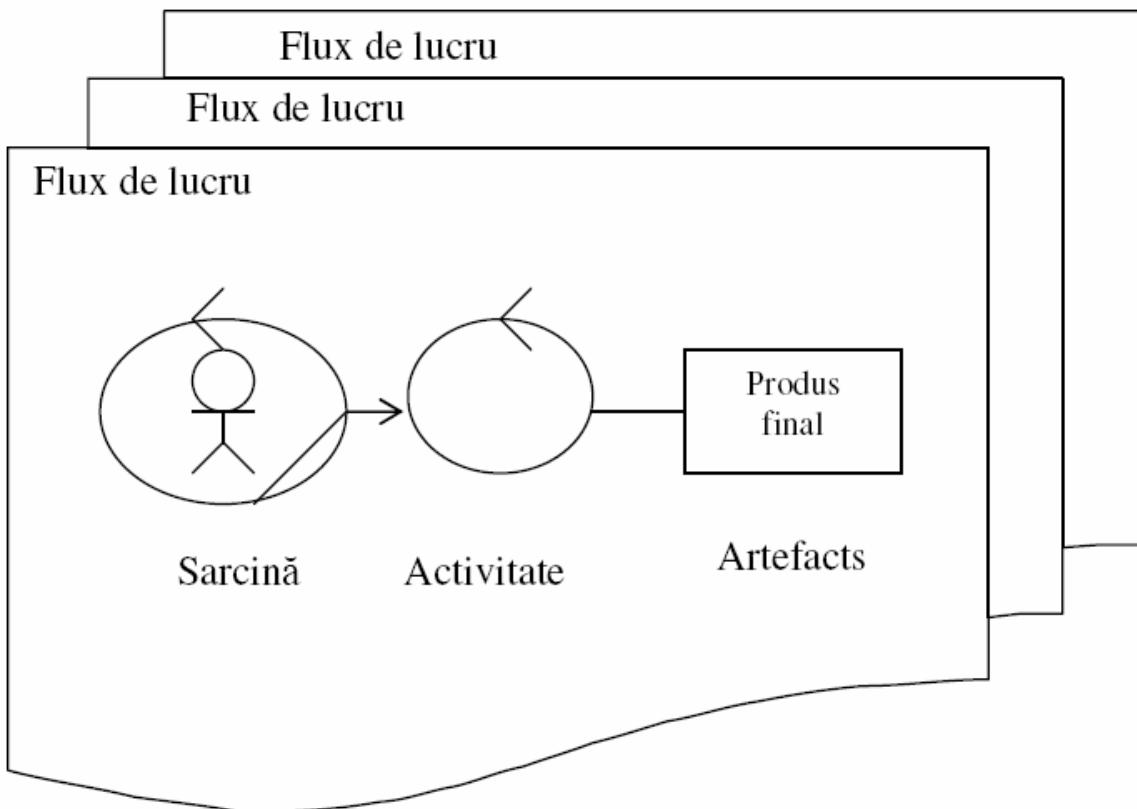
Evoluție istorică (3)

- (1996)
 - OOHDMD – Object-Oriented Hypermedia Design Method
- (1998) ← metode și tehnici pentru aplicațiile Web
 - WSDM – Website design Method
 - RUP – Rational Unified Process, metodologie unificata de realizare a sistemelor informatice
- (1999)
 - Unified Process – metodologie propusa de Jacobson, Booch si Rumbaugh
 - Multimedia Extension – metodologie bazata UML cu extindere în domeniul multimedia
 - Hypermedia Extension – tehnica bazata pe limbajul UML cu extindere prin profile pentru sistemele hypermedia
- (2000)
 - W2000 – Web 2000
 - WebML – Web Modeling language, tehnica de modelare a aplicatiilor Web
 - UWE – UML-based WebEngineering

Procesul unificat de dezvoltare a sistemelor de comerț electronic (Unified Process = UP)

- În domeniul abordărilor orientate-obiect UML a grupat cele mai bune soluții
- => proces de aplicare a UML-ului în dezvoltarea de sisteme informatiche (procesul **unificat de dezvoltare**)
- Concepte folosite:
 - Fluxuri de lucru (workflows)
 - Faze (activities)
 - Sarcini (workers)
 - Produse obținute (artefacts)

Conceptele utilizate



Ciclul de dezvoltare software

- 4 faze corespunzătoare perspectivei manageriale asupra procesului
- Studiul preliminar
 - Amplasarea viitorului sistem în cadrul organizației
- Elaborarea
 - Cerințele funcționale și non-funcționale ale sistemului
- Construcția
 - Analiza, proiectarea, implementarea, testarea
- Tranzită
 - Introducerea în exploatare a sistemului

Studiul preliminar

- Definirea amplasării viitorului sistem în cadrul activității organizației
- Delimitarea ariei de cuprindere
- Stabilirea obiectivelor
- Rezultatul fazei: “viziunea sistemului”

Elaborarea

- Precizează cerințele funcționale și non-funcționale ale sistemului
- Rezultate:
 - Arhitectura sistemului
 - Modelul detaliat al cazurilor de utilizare
- Modelul include:
 - Descrierea cazurilor de utilizare
 - Specificarea actorilor implicați
 - Eventuale cerințe non-funcționale
 - Prototipuri ale interfețelor cu utilizatorii

Construcția

- Asigură obținerea sistemului
- Include:
 - Analiza
 - Proiectarea
 - Programarea
 - Testarea
- Rezultatul:
 - Modelul de analiză și proiectare

Analiza

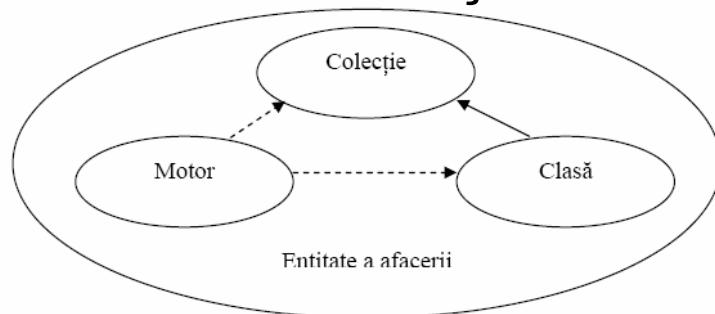
- Model de analiză ce cuprinde
 - Obiectele și clasele domeniului
 - Relațiile și comportamentul acestora
- Documentul arhitecturii software
- Sistemul este împărțit în subsisteme
- Pașii de urmat:
 - Identificarea obiectelor și a claselor
 - Generarea de realizări ale cazurilor de utilizare
 - Identificarea asocierilor și agregărilor
 - Definirea atributelor claselor
 - Identificarea relațiilor de moștenire
 - Definitivarea modelului analizei

Proiectarea (1)

- Transformarea modelului de analiză într-un sistem software
- Cuprinde
 - Adăugarea de noi clase pentru persistență
 - Revederea și definitivarea claselor, atributelor, operațiilor, relațiilor definite anterior
 - Extinderea și completarea laturii logice a arhitecturii
- Factori de influență:
 - Limbajul de programare
 - Mediul de dezvoltare
 - Stilul de arhitectură
 - Tehnologiile avute în vedere

Proiectarea (2)

- Pașii de urmat:
 - Proiectarea claselor
 - Identificarea și definitivarea interfețelor
 - Proiectarea realizărilor de cazuri de utilizare
 - Proiectarea subsistemelor
- Ex.: aplicarea şablonului motor-colecţie-clasă (MSDN)



Implementarea

- Obținerea programelor
- Transpune conținutul diagramelor și specificațiilor de proiectare în forma specifică limbajului de programare folosit
- UML ajută la obținerea:
 - Modelul de implementare
 - Modelul de distribuire și procedura de integrare a elementelor/componentelor sistemului

Testarea și Tranzită

- Testarea
 - Identificarea și corectarea eventualelor erori de programare
 - Sunt folosite cazuri de test (teste de integrare, de sistem)
- Tranzită
 - Introducerea în exploatare a sistemului
 - Rezultatul:
 - Versiunea finală a sistemului

Metodologia UWE

- UML-Based Web Engineering (UWE)
 - Metodologie centrată pe utilizatori, bazată pe UML, iterativă și incrementală, dezvoltată pentru realizarea sistemelor hipermedia
- Sistemul hipermedia
 - Reflectă caracteristicile utilizatorilor
 - Adoptă un model centrat pe adaptarea aspectelor vizibile ale sistemului la cerințele utilizatorilor

Diferențe

- Față de UP, UWE aduce următoarele:
 - Specializarea pe dezvoltarea aplicațiilor Web
 - Ciclul de viață include și faza de întreținere
 - 2 fluxuri suplimentare: managementul proiectului și managementul calității
 - Managementul calității include (pe lângă testare) și validarea cerințelor și verificarea proiectării
 - Propune stereotipuri noi (extensii UML-profile)
 - Include o metodă sistemică de analiză a aplicațiilor Web

Ciclul de viață al unui proiect în UWE

- Inițiere
- Elaborare
- Construcție
- Testare
- Întreținere

Întreținerea

- Fază suplimentară adaugată peste modelul UP
- Începe atunci când prima versiune a aplicației Web este livrată
- Se termină atunci când aceasta nu mai este folosită de nimeni
- În cadrul fazei sunt realizate:
 - Ajustări
 - Îmbunătățiri
 - Posibile noi dezvoltări ale aplicației

Fluxurile de lucru

- Stabilirea cerințelor
 - Analiza și proiectarea
 - Managementul riscului
 - Planificarea iterățiilor
 - Validarea
-
- Fiecare fază este urmată de un flux de iterății

Stabilirea cerințelor

- Participă experți ai domeniului de studiat, arhitecți, analiști și proiectanți de interfețe Web
- Fluxul începe cu identificarea utilizatorilor
- Se identifică cerințele funcționale (de conținut, de structură, de prezentare a informațiilor, de adaptare, de utilizator) și non-funcționale
- Furnizează un model al cazurilor de utilizare și un prototip de interfață utilizator

Analiza și proiectarea

- Cerințele anterior identificate devin specificații de descriere a modului de implementare
- Rezultate:
 - Conturarea arhitecturii sistemului
 - Modelul conceptual
 - Modelul utilizator
 - Modelul interfețelor
 - Modelul prezentării informațiilor
 - Modelul de adaptare
 - Proiectarea claselor, subsistemelor și interfețelor
- Analiza: cerințe funcționale
- Proiectarea: cerințe non-funcționale

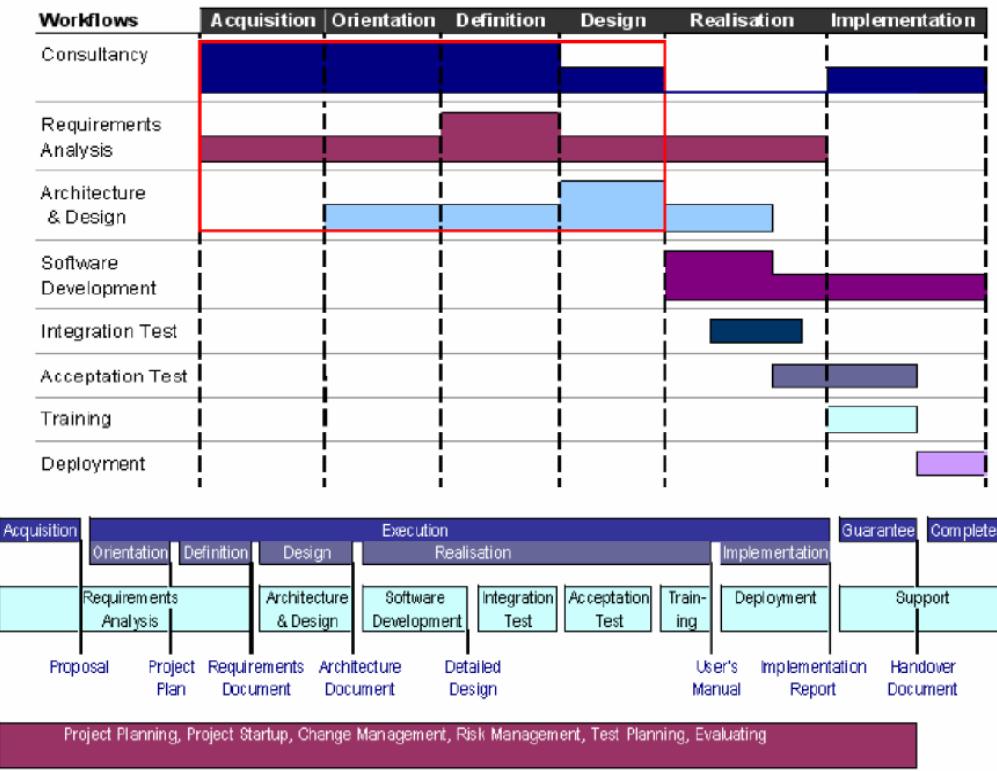
Alte etape

- Managementul riscului
 - Identificarea riscurilor procesului de realizare
 - Strategii de risc
- Planificarea iterățiilor
 - Stabilește starea inițială și finală a proiectului, costuri, rezultate, planul de livrare al proiectului
- Validarea
 - Verificarea rezultatelor obținute

Metodologia WEM

- WEM = Web Engineering Method
- Metodologie de proiectare pentru sisteme CMS-based Web
- Concepte împrumutate din ambele tehnologii anterior prezentate:
 - PU adecvat dezvoltării în componente
 - PU standard recunoscut și acceptat
 - UWE adoptă caracteristici specifice sistemelor Web
 - Ambele folosesc UML (standard în modelarea orientată obiect)

Descrierea metodologiei



Accounting
Project Phases
Workflows
Project Products
Project Management

Modelarea sistemelor informatice de e-Commerce

CUMPARA INTELIGENT DIN PRIMA RETEA DE MAGAZINE O...

Subject: CUMPARA INTELIGENT DIN PRIMA RETEA DE MAGAZINE ONLINE SPECIALIZATE.
From: "Newsletter i-shops.ro" <i-shops.ro@gmail.com>
Date: Thu, 02 Apr 2009 01:27:19 -0300
To: "ciprian.dobre@cs.pub.ro" <ciprian.dobre@cs.pub.ro>



CUMPARA INTELIGENT DIN SINGURA RETEA
DE MAGAZINE ONLINE SPECIALIZATE

e-lenjerie.ro



perfumery-shop.ro



e-notebook.ro



e-lcd.ro



e-imprimante.ro



e-licentesoft.ro



Am putut avea o-mail doar unde din urmatoarele motive: nu avem primirea acestor informații, suntem un client corporativ nou și, suntem în baza noastră de abonamente a unui corespondent acordat, altă cunoaștere de e-mail a fost detectată dincolo de baza de date la care nu subsecă, altă cunoaștere să fie făcută publică prin afara de canale publice sau ce regăsește în situri publice și a fost fricătoare pentru un altul în cunoașterea informațiilor noastre sau perioada

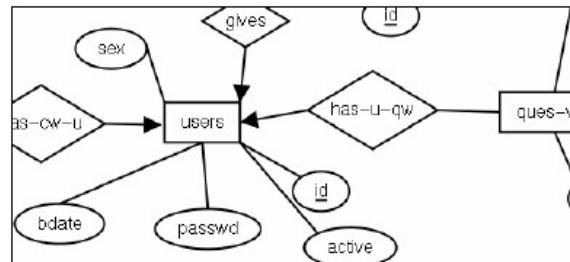
De ce modelăm?

- Aplicațiile software pot fi complexe; cum le putem gestiona eficient?
Posibilă soluție: folosim o bună reprezentare a sistemului
- **Model**: reprezintă anumite proprietăți ale unui obiect într-un anumit context
 - Reducerea complexității prin ascunderea detaliilor ce nu sunt necesare (abstractizare)
 - Anumite proprietăți ale sistemului pot deveni mai vizibile (claritate)
 - Facilitează aplicarea unei **metodologii**
 - În general; modelele oferă o mai bună **utilizabilitate**
- De obicei se folosesc mai multe modele în paralel; modele diferite pentru diverse scopuri

Modele

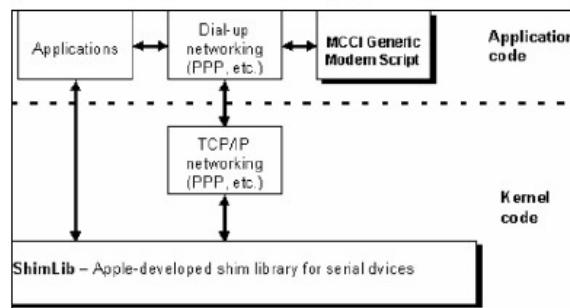
- **Modele de date**

Tipuri de date și relațiile între acestea
E.g. Diagrame ER, diagrame de clase



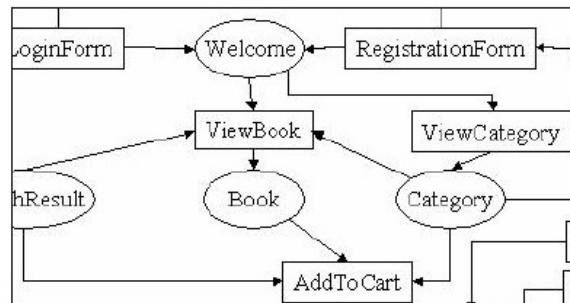
- **Modele arhitecturale**

Componențe ale unui sistem și relațiile între acestea



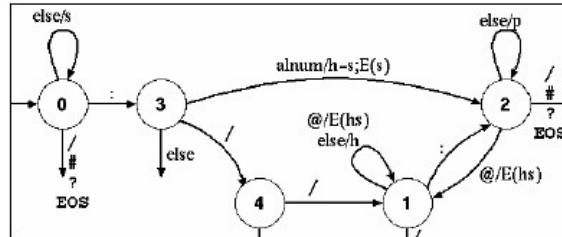
- **Modele pentru interfețe utilizator**

Structura UI
(navigare, interacții, ...)
E.g. formcharts, diagrame ecran

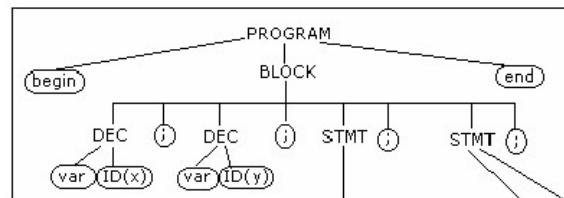


Mai multe modele

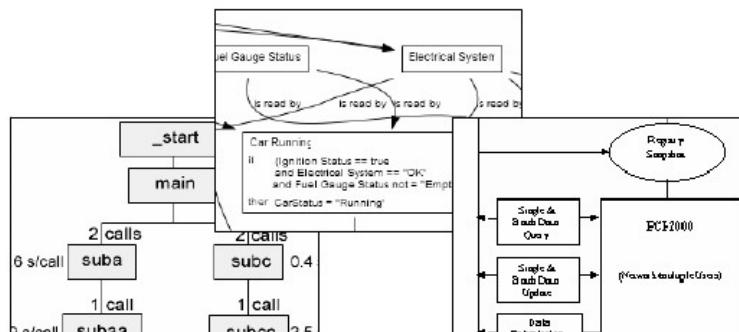
- **Modele stare tranzitie**
Stări ale sistemului și tranzitii
între acestea
E.g. Mașini cu stare pentru
modelarea jocurilor



- **Modele ale codului sursă**
Structura codului programului
E.g. Arboi de sintaxă abstractă
(AST)



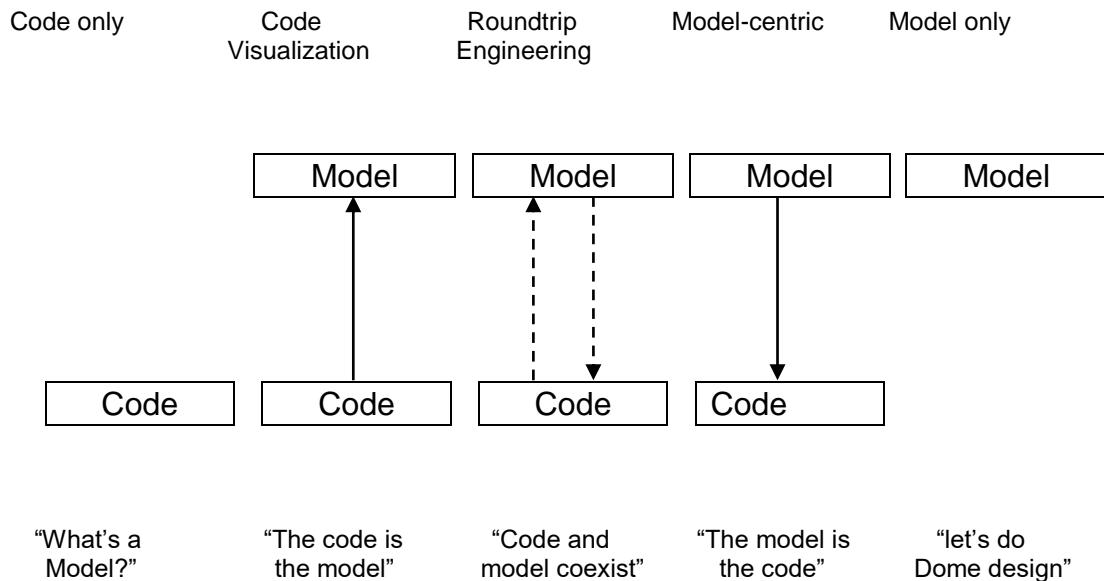
Grafuri de apeluri, grafuri de dependințe, diagrame de fluxuri de date
& multe altele...



Modelare orientată obiect

- Punct de pornire: descrierea informală a cerințelor
- **Analiza OO:**
 - Modelează invarianți specifici domeniului unui sistem
 - Părți stabile ce descriu concepte stabile
 - Ex: într-o universitate sunt întotdeauna studenți, cursuri, profesori...
- **Proiectarea OO:**
 - Rafinează modelul de analiză cu părți dependente de implementare
 - Ex: persistență, distribuție, folosirea anumitor tehnologii sau componente

Moduri de sincronizare modele-cod sursă aplicate



Evoluția sistemelor (1)

- Abordare code-only
 - sistemele sunt scrise direct într-un limbaj de programare
- Abordarea code visualization
 - după analiza problemei, codul e reprezentat grafic și modificat
- Dezvoltarea round-trip engineering – RTE
 - Separarea modelelor de codul sursă
- Abordarea centrată pe modele (model centric approach)
 - modelele sistemului informatic sunt suficient de detaliate pentru a permite implementarea completă a acestuia
- Abordarea model-only
 - modelele sunt folosite strict la înțelegerea și reprezentarea domeniului de studiat, a proceselor de afaceri, a analizei arhitecturii sistemului, etc.

Evoluția sistemelor (2)

- **Abordarea MDA**

- cea mai recentă abordare
- asemănătă mai mult cu o abordare în care codul este parțial sau complet generat pe baza mai multor modele, obținute prin aplicarea de diferite limbaje standard de modelare
- acesta este modul de dezvoltare de sisteme informaticice care poate răspunde cu acuratețe cerințelor beneficiarilor, oferind mai multă flexibilitate impusă de evoluția acestora în timp
- abordarea MDA permite dezvoltarea unui sistem IT pe baza standardelor existente deja, furnizând un cadru de interoperabilitate și interconectare a sistemelor software diferite.

Practica modelării sistemelor informaticice

- Modelele reprezintă o abstractizare aproximativă a elementelor reale ce urmează a fi realizate.
- Mai puțin la început pentru că aplicațiile software erau mai simple, ușor de realizat, dar mai ales pentru că puteau fi modificate fără costuri suplimentare semnificative
 - există tiparul **imaginează-construiește-modifică**
- Astăzi sistemele informaticice sunt din ce în ce mai complexe
 - se impune integrarea cu aplicații software deja existente
 - sunt utilizate permanent
 - se impune adaptarea lor la noile condiții reale.

Ce modele sunt importante?

- Business Model
 - Vizualizarea proceselor de business
- System Architecture Model
 - Vizualizarea cerințelor de sistem, structurii și comportamentului
- Use Case Model
 - Vizualizarea cerințelor funcționale
- Analysis Model
 - “Ce” trebuie să realizeze sistemul pentru a realiza cerințele funcționale impuse
- User Experience Model
 - Vizualizarea interacțiunii utilizatorului cu sistemul
- Design Model
 - “Cum” realizează sistemul cerințele funcționale
- Data Model
 - Vizualizarea stocării persistente
- Implementation Model
 - Vizualizarea codului

Abordarea IBM

- Beneficiarii produselor software aşteaptă livrarea la timp și de calitate a unui produs care să răspundă cerințelor lor.
- Specialiștii firmei IBM au formulat **patru caracteristici** pe care trebuie să le îndeplinească în procesul de dezvoltare software, și anume:
 - dezvoltare iterativă;
 - concentrare pe arhitectura sistemului informatic;
 - asigurarea continuă a calității impuse de beneficiari și gestionarea schimbărilor și rezultatelor obținute, cât și a complexității sistemului IT;
 - înțelegerea procesului de proiectare și a riscurilor asociate.

Abordarea IBM

- Prin modelarea sistemelor informaticice, dezvoltatorii software pot:
- crea modelele sistemului înaintea angajării de resurse adiționale;
- proiecta aplicații software pornind de la cerințe, dând siguranță și încredere în calitatea sistemului obținut;
- aplica dezvoltarea iterativă în care modelele sau alte artefacte obținute au un nivel ridicat de abstractizare, permitând modificări rapide și oricât de frecvente sunt necesare, funcție de modificările mediului de lucru.

Reticență la modelare

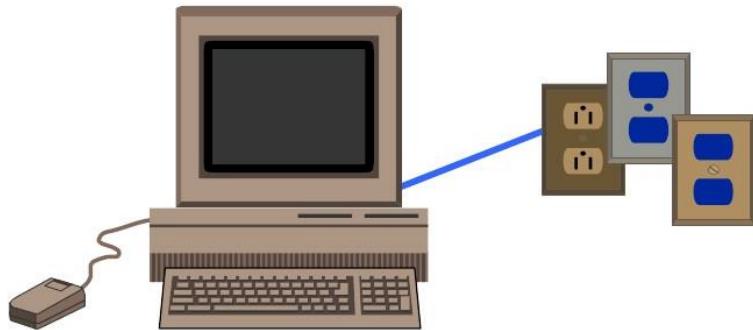
- Nu toți dezvoltatorii software înțeleg necesitatea modelării software
 - Adesea sistemele informaticice sunt simple, ușor de înțeles și abia apoi devin din ce în ce mai complexe în mod natural.
 - În unele cazuri ei nu apelează la modelare pentru că pur și simplu nu percep necesitatea acesteia decât mult mai târziu.
- Unii specialiști argumentează rezistență la modelarea software ca fiind o trăsatură de cultură, sau, pentru că aceasta presupune instrumente suplimentare, pregatire suplimentară, anumite costuri adiționale, o durată mai mare de timp necesar de alocat și eforturi suplimentare.

Avantajele modelării

- înțelegerea mai bună a problemei de rezolvat
- realizarea unui sistem informatic de calitate
- proiectarea și construirea arhitecturii sistemului
- crearea unei vizualizări a codului sursă sau a altor forme de implementare a acestuia.

- Totuși modelarea nu reprezintă "totul sau nimic", ea reprezintă doar o parte din procesul de dezvoltare software.

Viziunea OMG



The Global Information Appliance

Consensul modelării

- Eterogenitatea este pretutindeni
 - Nu există consens asupra platformelor hardware
 - Nu există consens asupra sistemelor de operare
 - Nu există consens asupra protocolelor de rețea
 - Nu există consens asupra limbajelor de programare
- Totuși trebuie să existe concens asupra modelelor, interfețelor și interoperabilității!



Model Driven Architecture

- Inițiativa Model Driven Architecture (MDA_{TM}) a grupului OMG urmărește **integrarea** bunurilor deja existente în software-ul dezvoltat
- MDA ajută la integrarea mixului de soluții deja existente și oferă o arhitectură pentru a suporta orice schimbări neașteptate viitoare
- Focusată pe integrarea aplicațiilor de tip “legacy”
- Asigură integrarea facilă a aplicațiilor COTS
- Modelele sunt testabile și simulabile

Ce este MDA?

- Procesul de dezvoltare software nu mai seamănă nici pe departe cu procesul clasic “waterfall”, ci constă într-o permanență extindere și rafinare a unei soluții parțiale deja existente, căreia, după un număr de iterații, î se adaugă plus-valoare din punct de vedere al beneficiarului
- Def: **O modalitatea de a specifica și construi sisteme**
 - Bazată pe modelarea folosind UML
 - Suportă între ciclul de dezvoltare: analiză, proiectarea, implementare, deployment, menenanță, evoluție & integrarea cu sistemele viitoare
 - Interoperabilitate și Portabilitate
 - Folosește standarde deschise
 - Costuri inițiale scăzute și maximizarea ROI
 - Se aplică direct în mixul pe care astăzi îl avem în față:
 - Programming language
 - Operating system
 - Network
 - Middleware

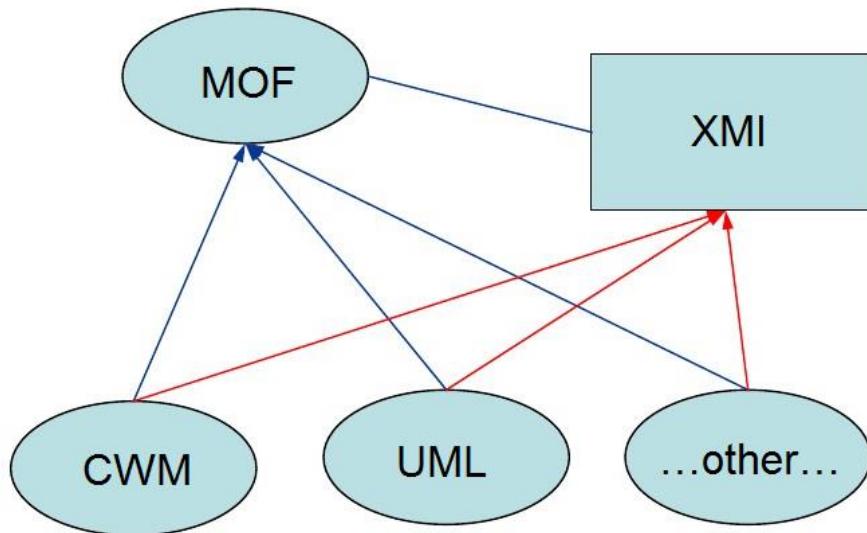
Categorii de instrumente MDA

- Instrumentele MDA se împart în trei categorii:
 - comerciale sau open source;
 - parțial sau complet de implementare;
 - generatoare de cod din modele sau executarea modelelor.

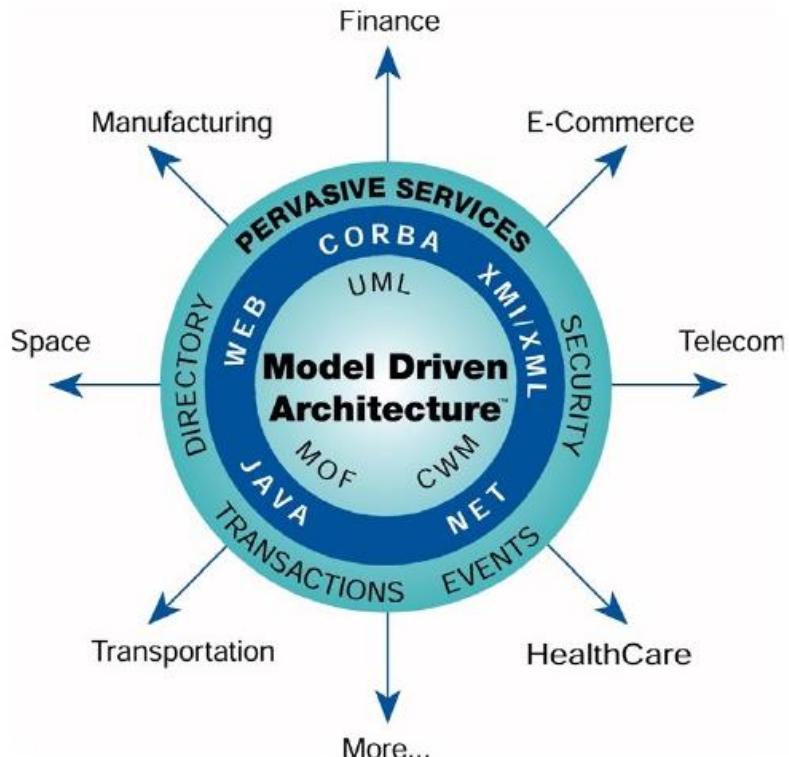
Suita de modelare a OMG

- **Unified Modeling Language**
 - UML™ rămâne singurul limbaj standardizat de modelare orientat obiect, cea mai bine cunoscută parte a standardului
- **Common Warehouse Metamodel**
 - CWM™, integrarea de inițiative data warehousing
- **Meta-Object Facility**
 - MOF™, standard de integrare a metadatelor
 - Definește metadatele și serviciile de metadate
- **XML Metadata Interchange**
 - XML™, standardul XML-UML
 - Interoperabilitate a instrumentelor UML
 - Colecție de reguli de mapare XML/MOF

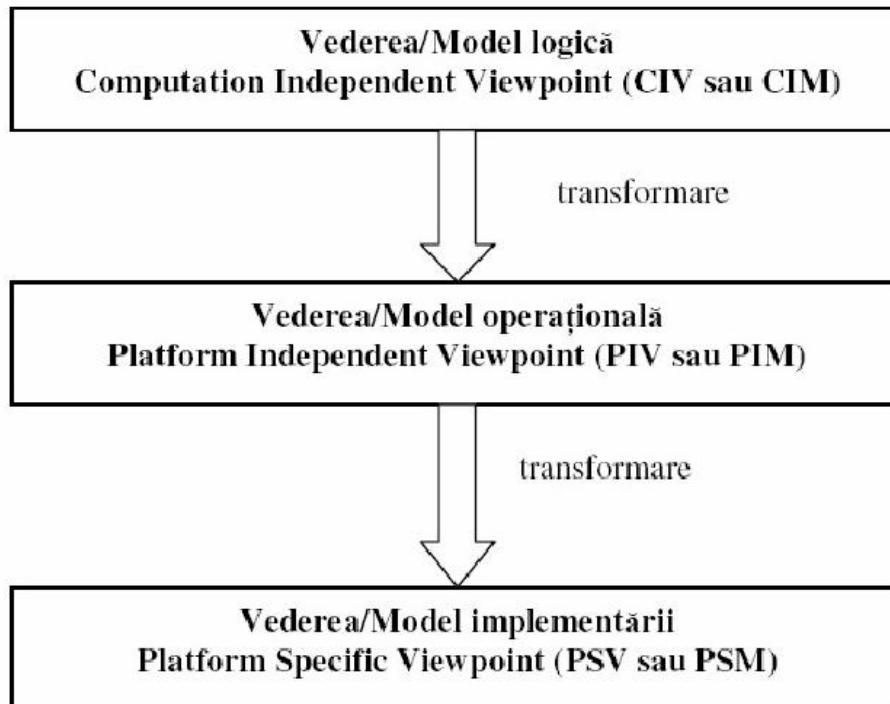
OMG Modeling Suite



Model Driven Architecture



3 vederi ale arhitecturii MDA



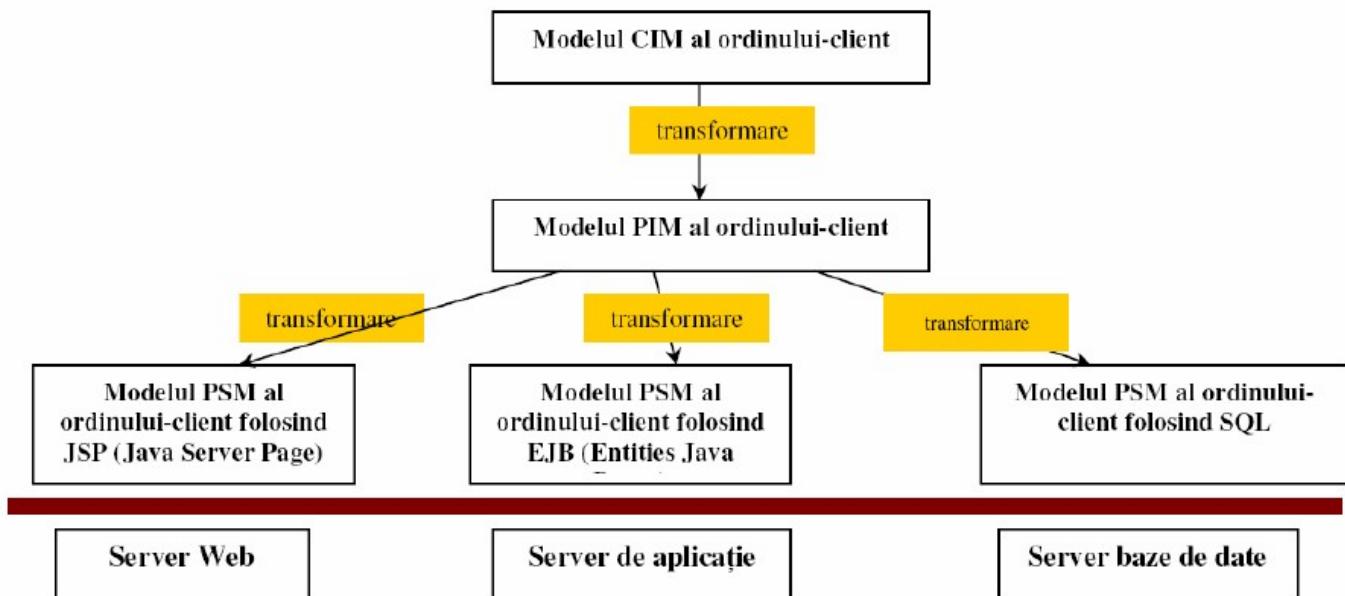
Vederi MDA

- Computation Independent Viewpoint – CIV
 - separă modelarea logică a sistemului informatic de specificațiile de implementare;
- Platform Independent Viewpoint – PIV
 - se focalizează pe modelarea operațională a sistemului, dar fără detalierea specificațiilor de implementare;
- Platform Specific Viewpoint – PSV
 - conține specificațiile de implementare (platforma hardware, platforma software, produse middleware, tehnologii IT folosite).

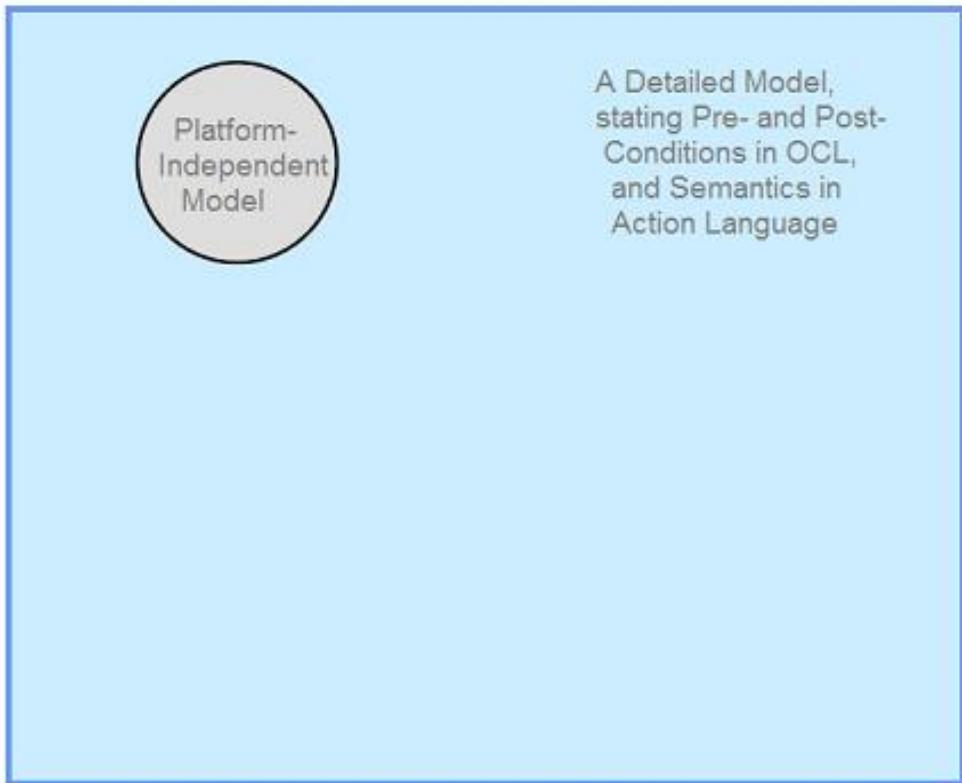
Exemplu (1)

- integrarea în sistemul informatic al unei firme a ordinelor de la clienți transmise prin Interne
 - vederea CIM va consta din diagrame UML la nivel conceptual prin care se arată ce va face sistemul informatic
 - vederea PIM descrie funcțiile și structura acestuia prin diagrame UML detaliate
 - mai multe vederi PSV

Exemplu (2)

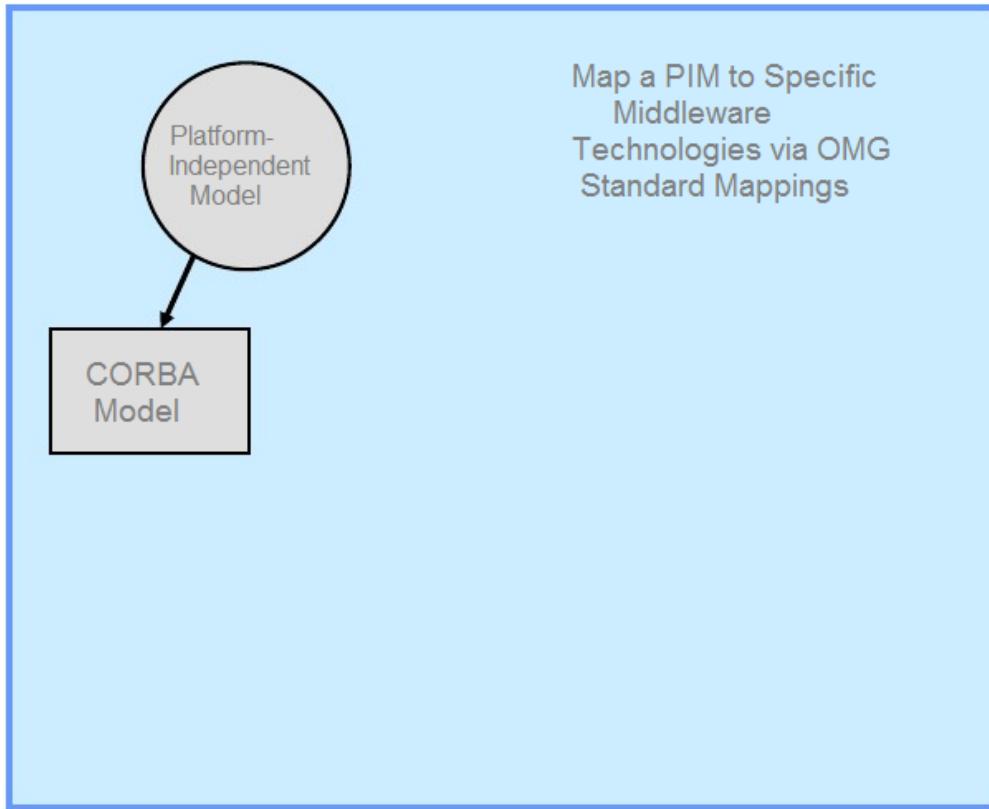


Construirea unei aplicații MDA



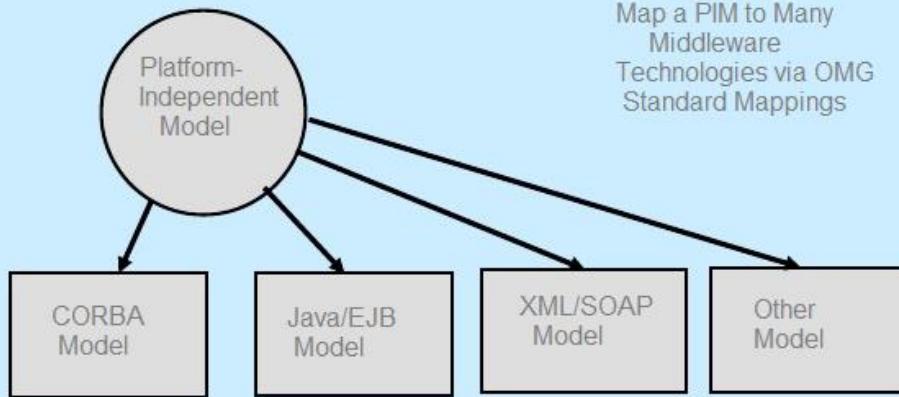
Start with a Platform-Independent Model (PIM) representing business functionality and behavior, undistorted by technology details.

Generarea modelului specific platformei



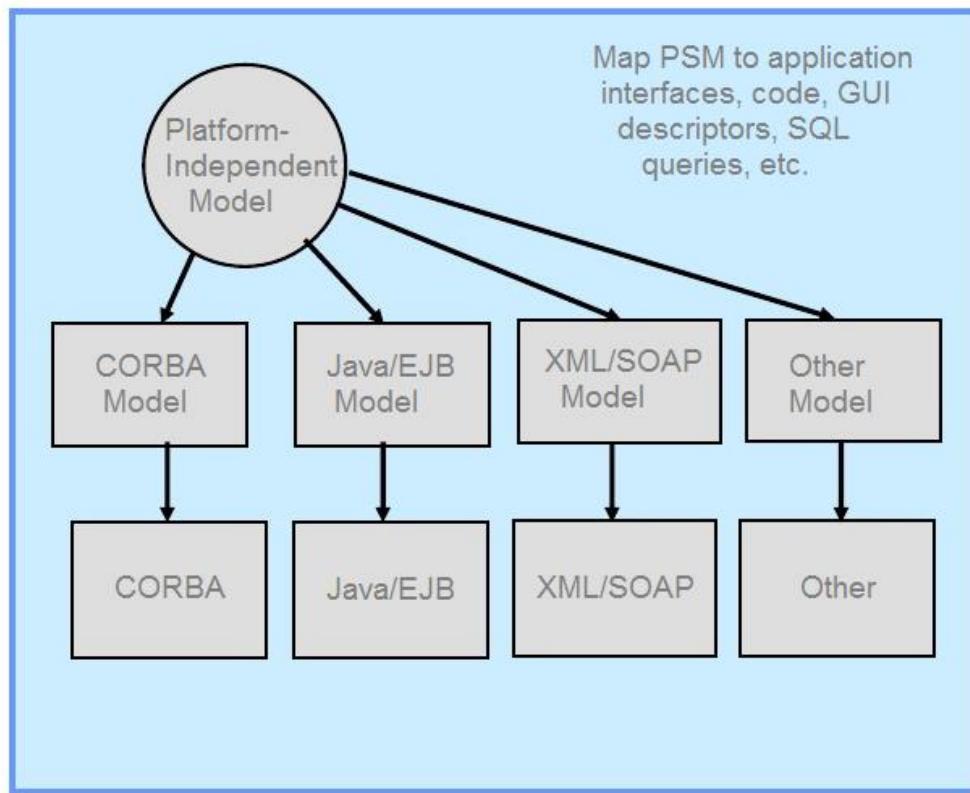
MDA tool applies a standard mapping to generate Platform-Specific Model (PSM) from the PIM. Code is partially automatic, partially hand-written.

Maparea pe multiple tehnologii de deployment



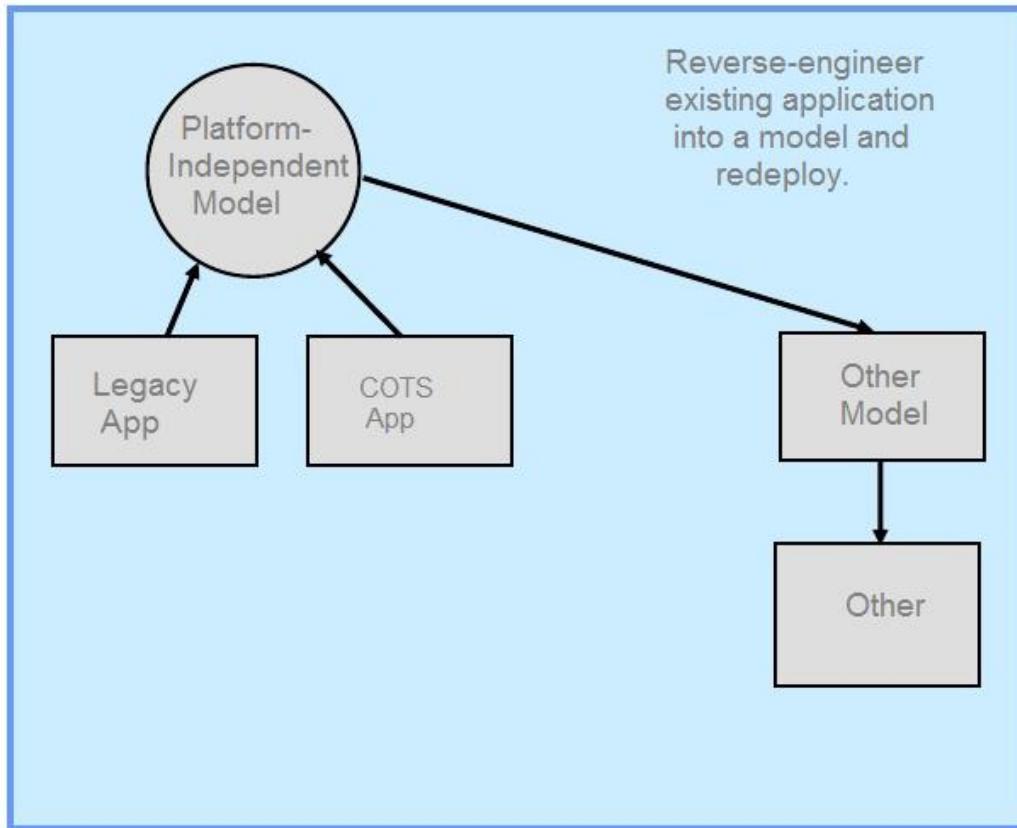
MDA tool applies a standard mapping to generate Platform-Specific Model (PSM) from the PIM. Code is partially automatic, partially hand-written.

Generarea implementărilor



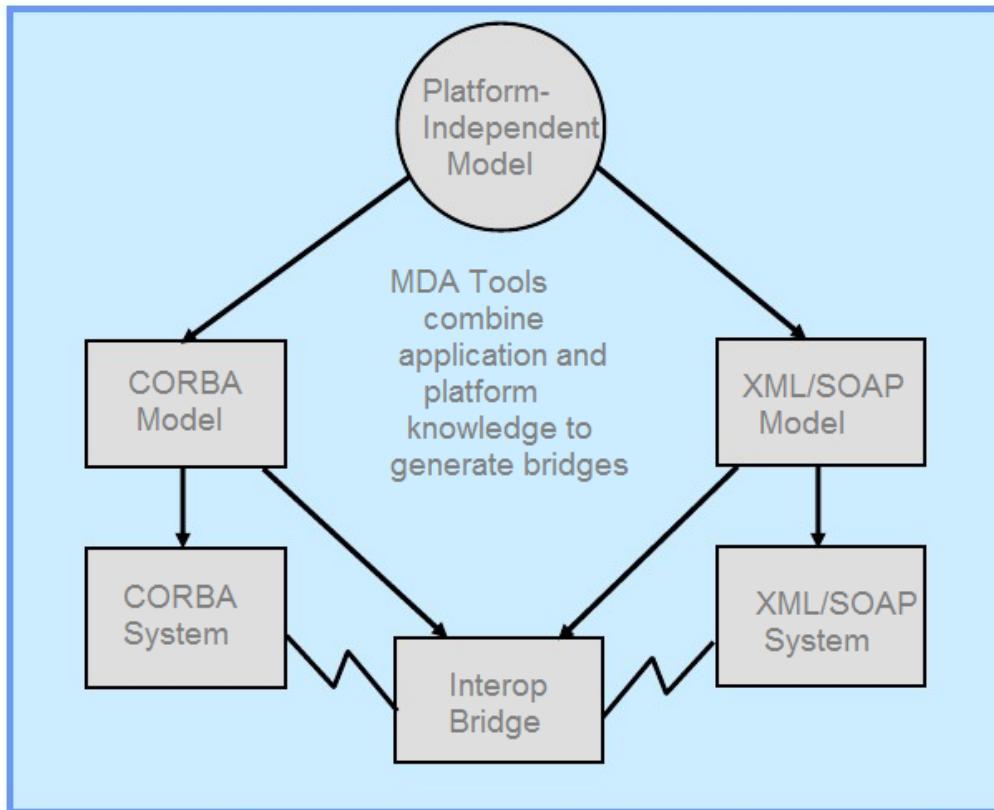
MDA Tool generates all or most of the implementation code for deployment technology selected by the developer.

Integrarea Legacy & COTS



MDA Tools for reverse engineering automate discovery of models for re-integration on new platforms.

Automating Bridges



Bridge generation is simplified by common application models, simplifying creation of integrated applications both within and across enterprises.

MDA în Standarde Industriale

- MDA promovează standarde ce sunt funcționale independent de tehnologie
 - Aplicabil pentru deploymenturi mari & mici, noi aplicații, legacy și COTS
 - Aplicabil pentru CORBA, DCOM, .Net, etc.
- MDA a fost adoptat de grupurile de standardizare OMG
- Grupuri din marketing văd valoarea adusă de abordarea MDA:
 - Legacy Transformation
 - Financial Services
 - Healthcare
 - Life Sciences Research
 - Manufacturing
 - Space & Ground Systems
 - Telecommunications

MDA În Practică

- Several excellent proofs-of-concept:
 - Wells Fargo (an architecture that has already been resilient through a decade of change)
 - Lockheed Martin Aeronautics
 - GCPR in US government
 - Deutsche Bank Bauspar
 - Defense Information Systems
 - Merrill Lynch
 - Österreichische Bundesbahn
 - Thales Training & Simulation
 - Zuercher Kantonal Bank
 - CGI
 - Chubb and Son

Mai multe informații

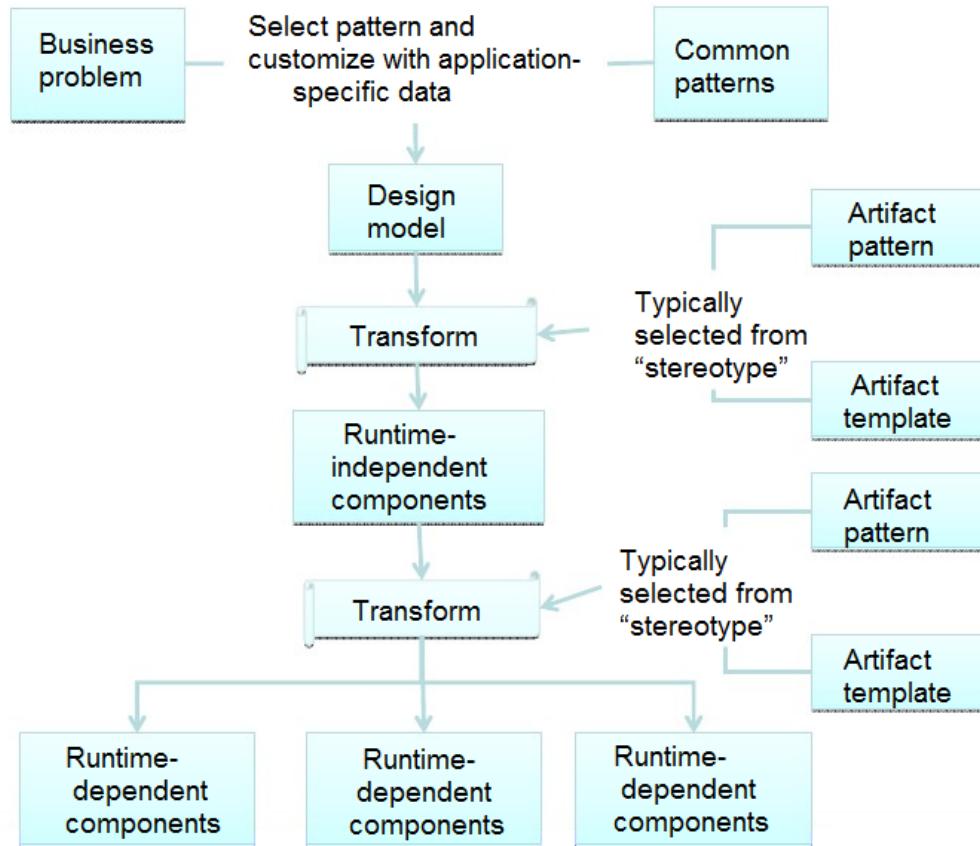
- MDA Information Page
 - <http://www.omg.org/mda/>
- OMG General Information
 - <http://www.omg.org/>

<http://www.omg.org/~soley/oois03.ppt>

MDD

- **Model Driven Development** (MDD) = dezvoltare dirijată prin modele
 - metodă de proiectare a sistemelor informaticе aplicabilă în cadrul metodologiei MDA.
- Se construiește un ansamblu de modele ale sistemului de analizat cât și ale noului sistem, pe baza cărora se generează alte modele sau codul sursă al sistemului.
- Totul se centrează pe transformarea modelelor sistemului de realizat și generare de cod sursă.
- Această metodă necesită un mediu integrat de dezvoltare (IDE) care să suporte:
 - limbajul UML, şabloane, transformarea modelelor UML și generare de cod sursă.
- **Rational Software Architect** (RSA) este un astfel de instrument integrat de proiectare și dezvoltare de sisteme informaticе.

Transformarea problemă reală – soluție IT aplicând metoda MDD



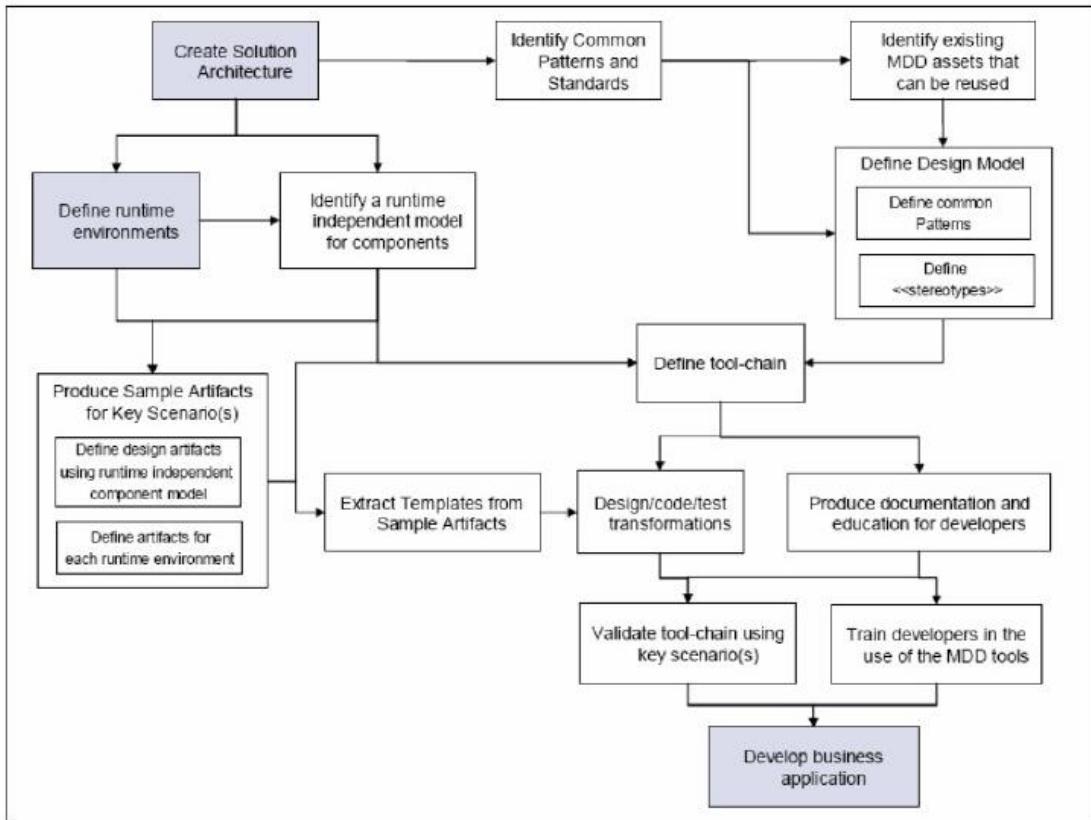
Şabloane

- Şabloanele sunt modalități de descriere a motivărilor, a ceea ce se va întâmpla și de ce.
- Motivele studierii și aplicării acestora sunt:
 - reutilizarea,
 - stabilirea unei terminologii comune ce îmbunătățește comunicarea echipei software,
 - trecerea la o perspectivă de nivel înalt asupra problemei de rezolvat,
 - îmbunătățirea posibilității de modificare a codului sursă,
 - facilitarea adoptării unor alternative de proiectare mai bune,
 - ofera alternative la ierarhiile laborioase de clase pentru probleme complexe cât și posibilitatea unei proiectări mai bune și nu numai a uneia funcțională.

Şabloane

- Integrarea şabloane-MDD
 - creşte productivitatea procesului de dezvoltare de aplicaţii soft
 - se îmbunătăşeşte calitatea sistemului livrat şi întreţinerea acestuia,
 - reutilizarea componentelor sistemului informatic,
 - reducerea costurilor şi a riscurilor unui proiect de investitii,
 - creşte flexibilitatea sistemului dezvoltat,
 - se îmbunătăşeşte comunicarea în cadrul echipei de dezvoltare cât şi dintre aceasta şi beneficiarii sistemului informatic.
- Metoda MDD se fundamentează pe abstractizare
 - permite realizarea vederii conceptuale a sistemului informatic, centrarea pe funcţiunile sistemului fără specificaţii de implementare.
- Este recunoscut faptul că este mai rapidă crearea modelelor unui sistem informatic decât scrierea de cod-sursă, motiv pentru care se agrează această metodă care permite lucrul la un nivel ridicat de abstractizare, permite includerea specificaţiilor de proiectare în modele, urmat de generarea codului sursă prin aplicarea de transformări automate între modele.

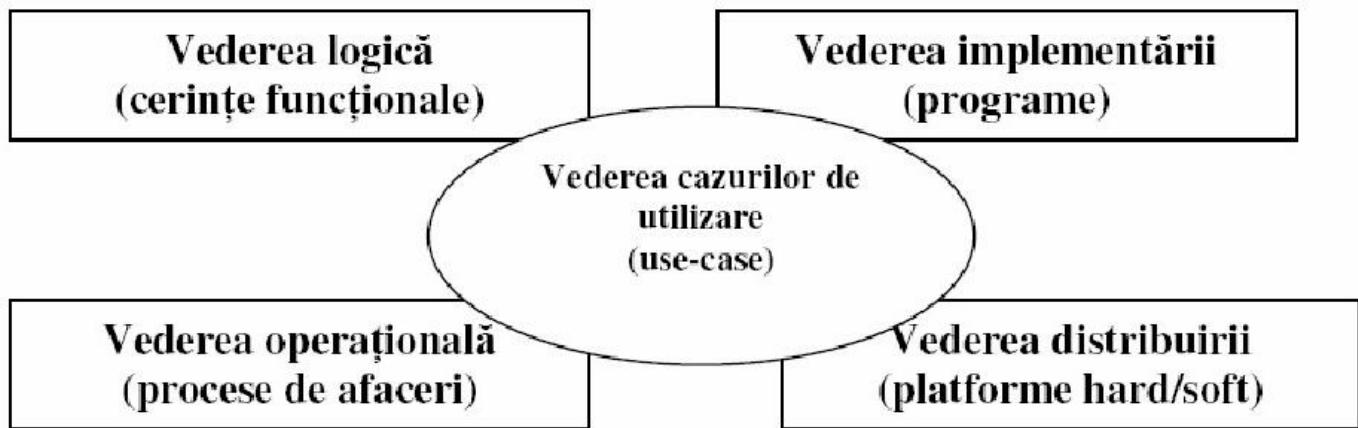
Etapele de realizare a sistemelor informatice în cadrul metodei MDD



Etape MDD

- Primele etape sunt de creare a arhitecturii sistemului și de definire a mediului de implementare (platforme soft/hard, tehnologii IT).
- Pe baza arhitecturii sistemului informatic sunt identificate şablonanele de analiză ce se pot aplica, cât și a modulelor de aplicații dezvoltate în alte proiecte MDD.
- Pe baza acestora se creează modelul de proiectare al sistemului și modelul de implementare pe componente, pe baza căruia se produc un număr de artefacte inițiale ce vor reprezenta scheme pentru următoarele transformări MDD.
- Se definesc instrumentele ce vor fi utilizate și o planificare a întregului proiect de dezvoltare a sistemului informatic.
- Sunt generate modele de programe (template-uri) pe baza artefactelor obținute anterior, care sunt testate, evaluate, sunt create module funcționale pentru care se furnizează și documentația aferentă.
- În acest moment se poate trece la dezvoltarea aplicațiilor, precedată de instruirea în prealabil a echipei de dezvoltare.

Perspective arhitecturale MDD

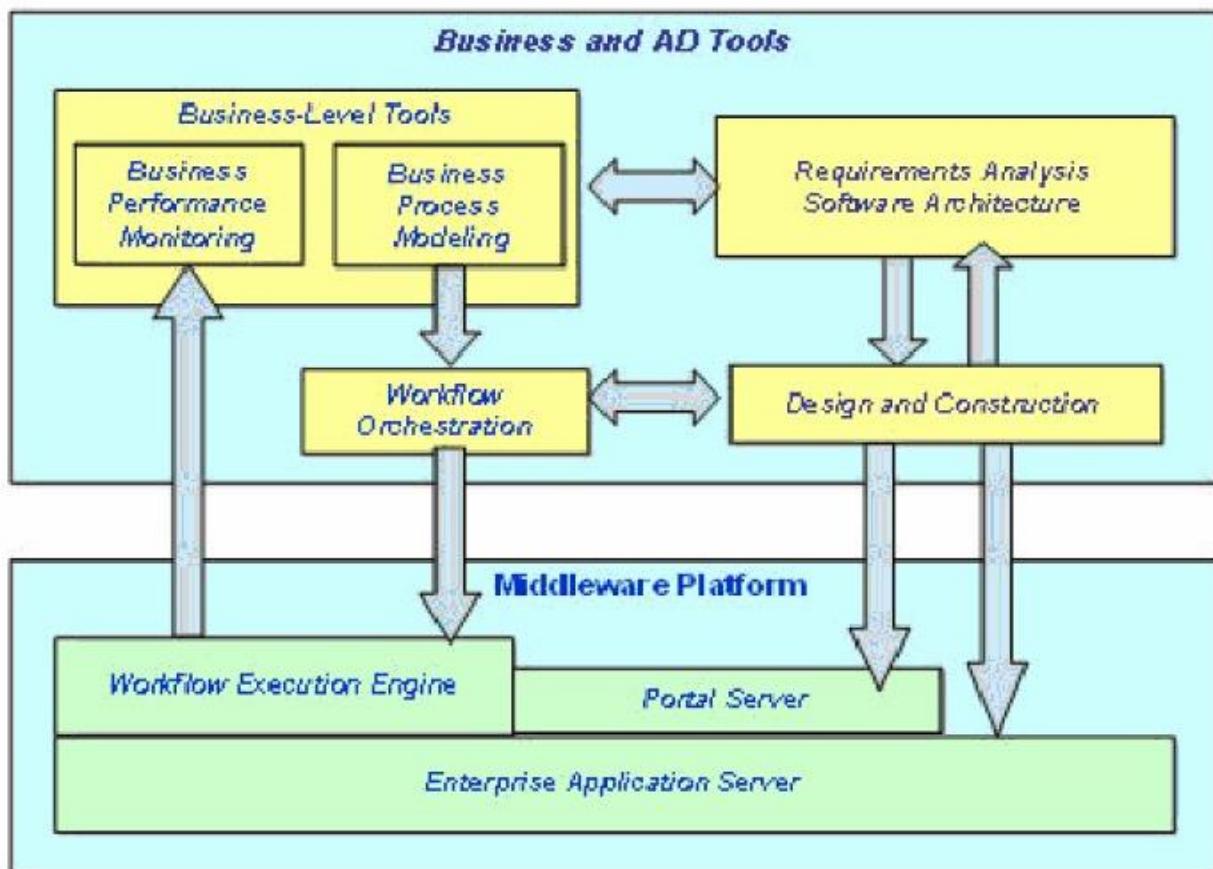


- vederea implementării - corespunde structurării programelor ce formează sistemul în componente
- vederea logică - descrie cerințele funcționale ale sistemului
- vederea distribuirii - definește aspectul spațial al sistemului (echipamente hardware, noduri de retea)-
- vederea operațională sau a proceselor, ce corespunde structurii de exploatare a programelor și componentelor executabile.

Analiza comparativă a procesului tradițional de dezvoltare a sistemelor informatiche și cel bazat pe metoda MDD

Dezvoltarea tradițională a sistemelor informatiche	Dezvoltarea sistemelor bazată pe metoda MDD
Centrat pe codificare și testare	Centrat pe analiză și proiectare, modelare
Codificarea manuală a programelor	Generarea automată a codului
Generarea manuală a documentației	Generarea automată a documentației
Testarea software-ului continuu	Validarea automată
Specificațiile sunt bazate pe hârtie	Prototipizare interactivă rapidă

Noua viziune privind instrumentele de proiectare și realizare de soluții IT



Componente în noua viziune

- **Platforma middleware** – reprezintă o platformă de execuție (runtime) formată din aplicații server robuste capabile să administreze soluții IT dezvoltate pentru platforme software eterogene sau soluții IT noi bazate pe standarde open;
- **Model de execuție** – interfețele platformei middleware definesc un model de programare ce trebuie înțeles de arhitecții IT. Orice utilizator al platformei middleware trebuie să aibă deja modelul conceptual al soluției IT și proiectarea fizică pe componente executabile;
- **Fluxurile de lucru** – reprezintă procesele de afaceri cât și relațiile dintre acestea și posibilități de configurare;
- **Instrumente de modelare a proceselor de afaceri** – includ instrumente de modelare a acestor procese cât și posibilitatea monitorizării acestora prin simularea impactului asupra evenimentelor declanșate în platforma middleware;
- **Arhitectura sistemului prin componente** – include instrumente care să permită gestionarea de aplicații moștenite, aplicații noi dezvoltate și asamblarea lor.

Rational Software Architect

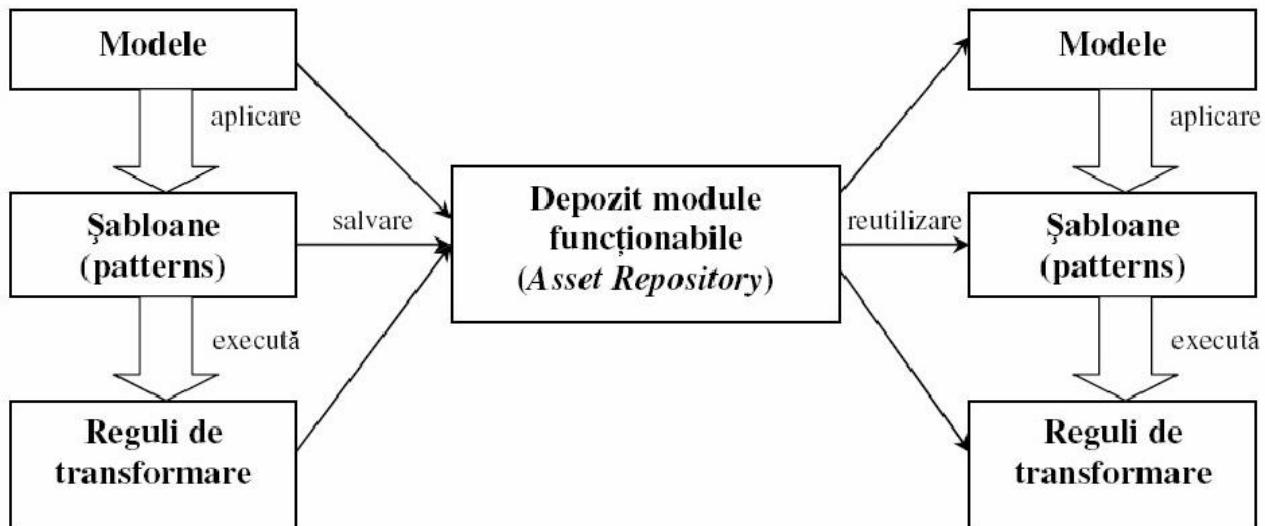
- octombrie 2004 - firma IBM a lansat platforma **IBM Software Development Platform** introducând o nouă generație de instrumente ce permit aplicarea metodei MDD.
 - Un loc central îl ocupă instrumentul integrat **Rational Software Architect** (RSA) ce suportă această metodă bazată pe limbajul UML.
- Instrument CASE ce se bazează pe metodologia MDA și oferă suport în dezvoltarea aplicațiilor Web statice sau dinamice ce pot rula pe platforma J2EE, include capabilități JSF, de utilizare a şabloanele (depozitul RAS – Reusable Asset Specification).

Obiectivele RSA

- separarea proiectării de implementare
- reducerea timpului și costului de proiectare și dezvoltare a sistemelor informaticе
- simplificarea și îmbunătățirea procesului de testare
- integrarea activităților de proiectare și dezvoltare
- standardizarea acestui proces
- creșterea calității sistemului realizat
- realizarea automată a unei documentații de calitate
- reutilizarea modulelor aplicațiilor și a documentației
- specificarea corectă și completă a cerințelor sistemului
- Simplitate
- flexibilitate.

Componentele mediului RSA

- La arhitectura mediului integrat RSA se adaugă componente specifice aplicării şabloanelor



IBM Rational Software Architect

An integrated platform for innovation and collaboration

Best of breed, comprehensive modeling tools that facilitate communication and collaboration

With the power of abstraction, automation and simplification

DoDAF

UML Profile-based Integrated Architecture

Profile for Software Services

Leveraging Jazz platform integrated with Rational Team Concert

Exploit the latest in modeling language technology and leverage an open and extensible modeling platform

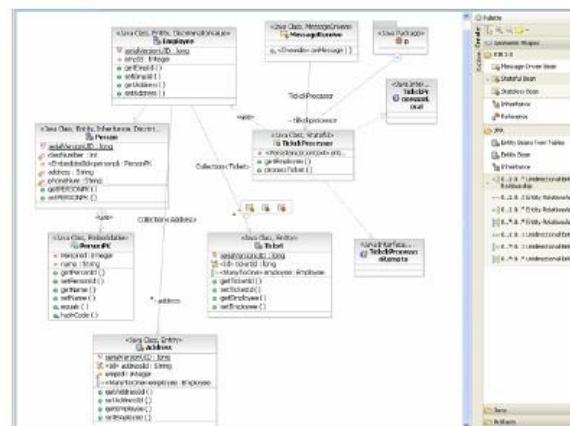
Simplify and unify Java and C++ design and development by integrating with other facets of the lifecycle such as:

Rational Data Architect, Rational Requisite Pro

Rational Asset Manager, Rational Team Concert

Rational ClearCase, Rational ClearQuest

Telelogic Synergy and Change



What's New: Rational Software Architect

Custom modeling environment for your business

Modeling with Domain Specific Language (DSL)

Work with reduced subset of UML

Rational Deployment Architecture Platform

Rich tools for deployment architecture definition

Verification tools for deployment architecture

Enhanced transformations and visualizations

Extensive Java and C++ support

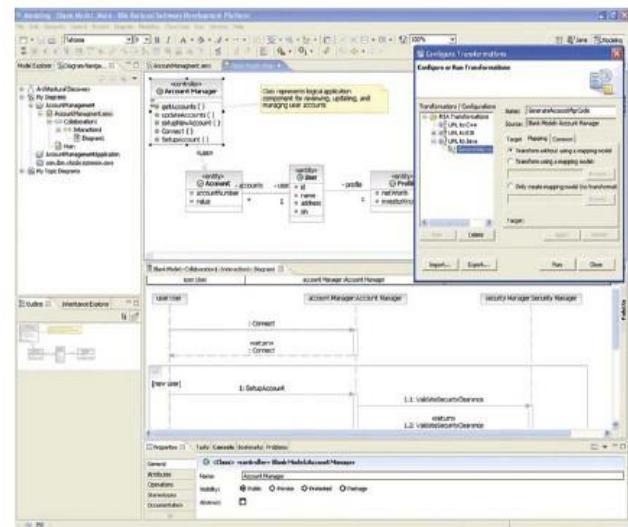
Increased visibility into existing source code

Integrations

Rational Asset Manager

Rational Team Concert on Jazz

Telelogic Change and Synergy



Getting Started: User Assistance

User Assistance model to enable users of all skill levels

Leverages Product Tours to assist with the discoverability of capabilities

Tutorial Gallery leverages tutorials as learning aids

“Watch and Learn”

“Play and Learn”

“Do and Learn”

Samples gallery provides completed assets for reference purposes

Showcase

Application

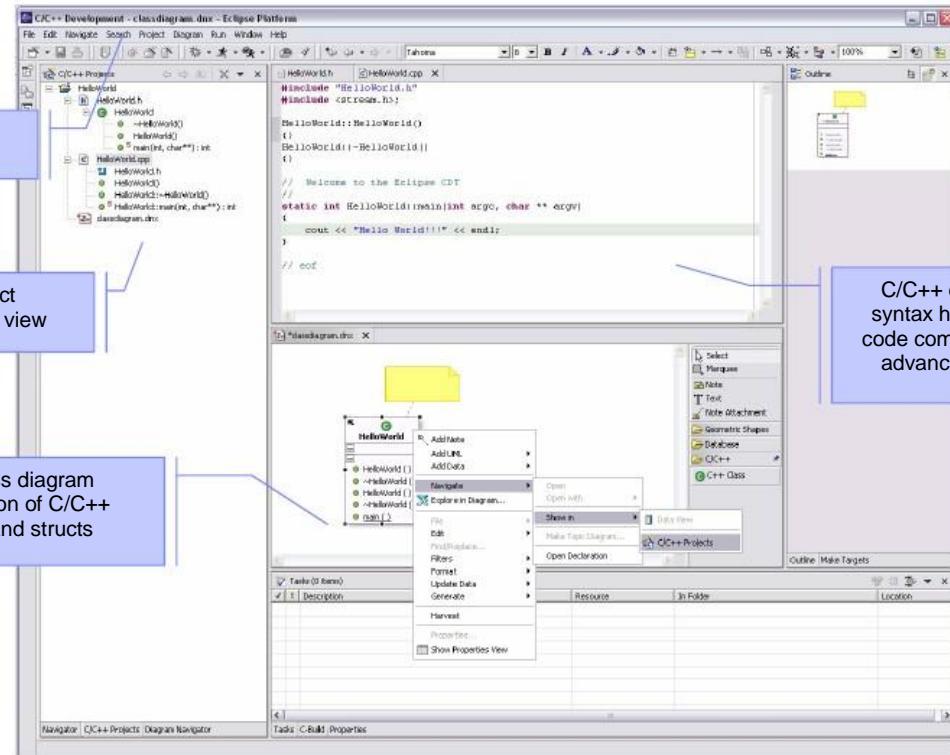
Technology

All user assistance can be launched from a Welcome perspective

The screenshot shows the Rational Software Development Platform's Welcome screen. At the top, there are five icons representing different capabilities: a globe, a footprint, a document with numbers, three circles, and a star. Below the icons are navigation links: Overview, First Steps, Tutorials, Samples, What's New, and Web Resources. A sub-section titled "Modeling software with UML" is displayed, featuring a icon of a document with a gear, a briefcase, and a starburst, followed by a description of architectural specification using UML. Another section titled "Developing the application code" shows an icon of a person working at a computer, a briefcase, and a starburst, with a description of application development for Java, C++, J2EEWeb, Web services, XML, and Data applications. A third section titled "Pattern and anti-pattern detection" shows an icon of a document with a gear, a briefcase, and a starburst, with a description of manually exploring application architecture using browser diagrams and automated discovery of patterns. The final section titled "Best practices and process" shows an icon of a person working at a computer, a briefcase, and a starburst, with a description of the Rational Unified Process configuration for software architects.



Key Feature: C/C++ Development Environment



Key Feature: C/C++ Development Environment

Editing and Navigation

C/C++ Syntax Highlighting, Outline View

C++ Class Browser (Hierarchy View)

C/C++ Search

C/C++ Content Assist

Project Import

Automated assistance in setting up CDT for search and content assist.

UML C/C++ Code Editor

Debug

GDB Integrated

Extensible Debug Interface

Build

Standard Make for projects with existing build infrastructure

Managed Build

Automatic makefile generation

GNU tools supported out of box

Managed build is extensible, build tools can be plugged-in and build tools options selectable

Meets Internationalization and Accessibility requirements

Extensibility

Provides extension points for managed build, debuggers, ...



Key Feature: Modeling assistance

Simplify the capture of UML models during Analysis and Design

Make modeling more accessible to a broader audience

New custom views improve the editing experience

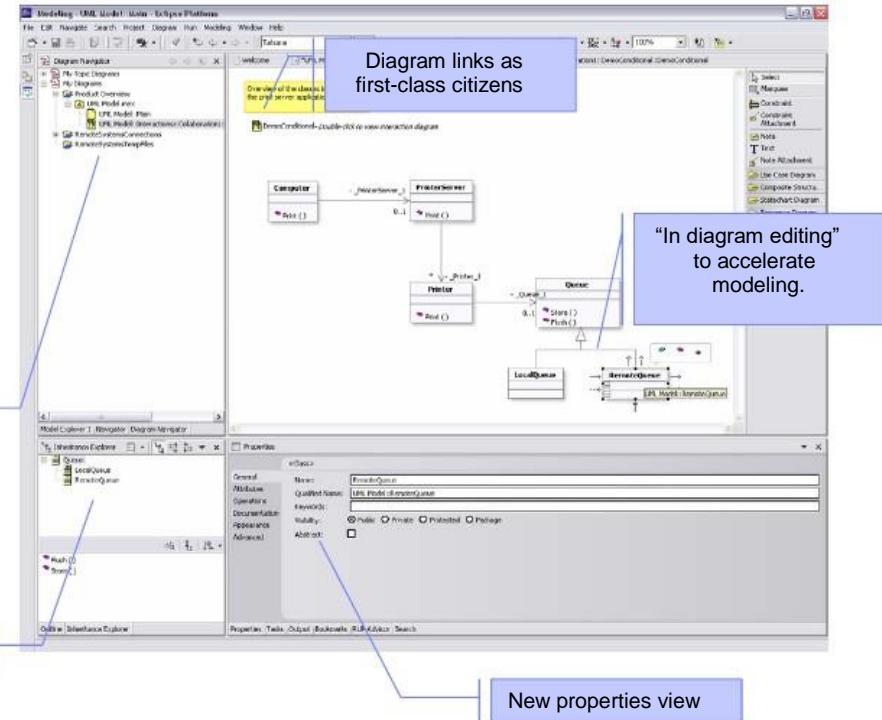
New "Diagram Navigator" view provides a diagram filtered view of the models and workspace

Inheritance view

Diagram links as first-class citizens

"In diagram editing" to accelerate modeling.

New properties view



Key Feature: Patterns

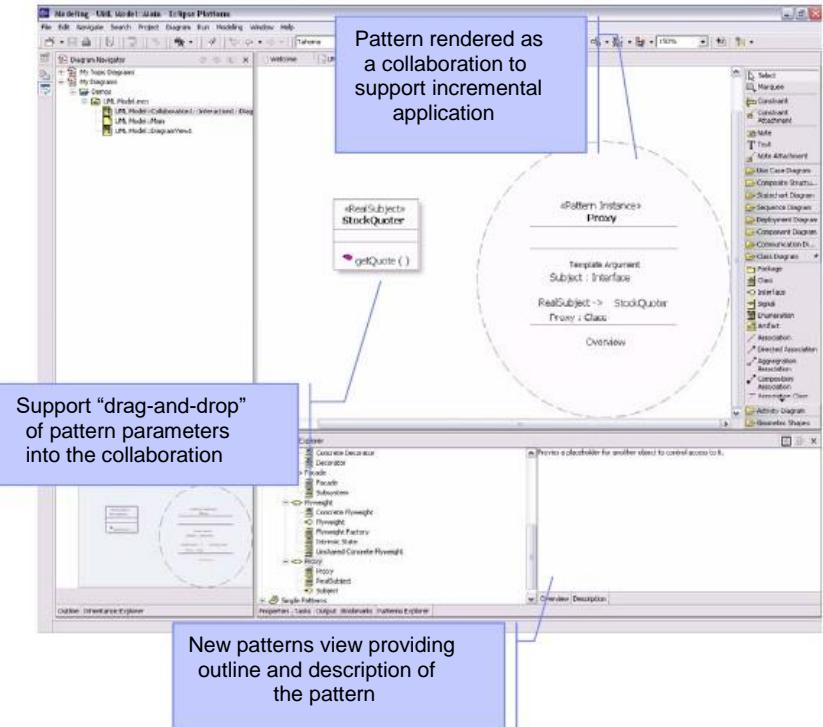
Applying Patterns is very simple

Evolution of pattern experience
based on lessons learned

Pattern-authoring provides greater
flexibility using Open API

All Gang of Four design patterns
provided

Additional patterns provided via
RAS repository on IBM
developerWorks



Key Feature: Transformations

Transformations are optimal for “batch” style computationally intensive operations

Model-to-model

Model-to-code

Code-to-model

Out-of-the box code transforms

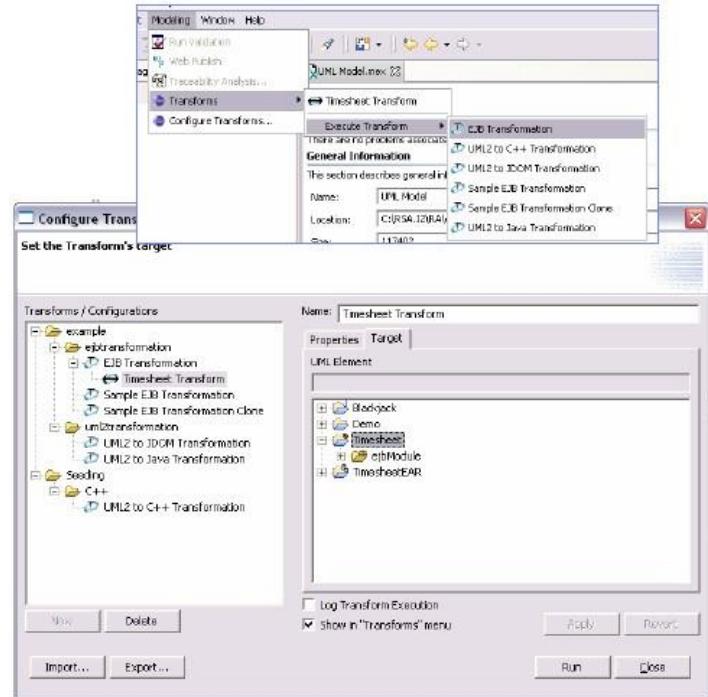
UML-to-Java/JSE

UML-to-C++

UML-to-CORBA IDL

Plus sample model-to-model transforms

Transformations may be updated via RAS repository hosted on IBM developerWorks



Key Feature: Architectural Analysis, Discovery and Control

Architecture discovery for J2SE

High-level software visualization

Application architecture is reflected in the running code

Analyzing code can help assess its maintainability

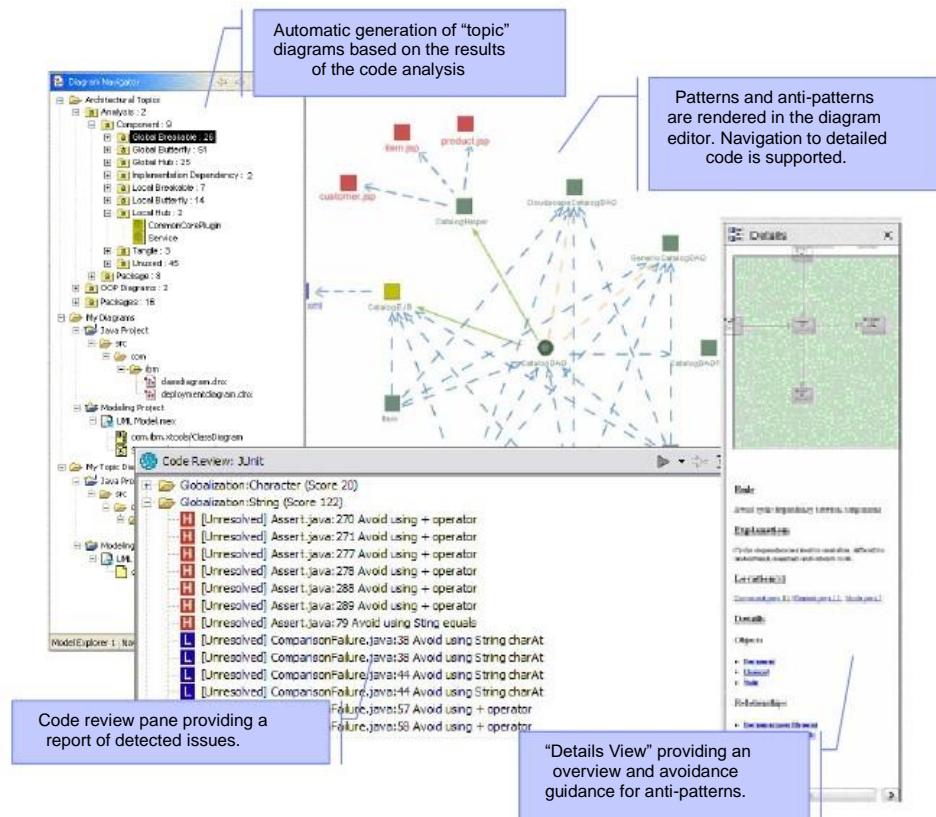
Govern the architecture with the assistance of rules

Template-based rule authoring

Anti-pattern and pattern detection

Detection of cyclic dependencies,
hubs, breakable, etc.

Wizard assisted automated quick-fix



Key Feature: Visualize method bodies

Facilitates understanding and application's behavior by providing visualization of detailed code

Diagrams can be integrated in Javadoc reports

The screenshot displays the Rational Application Developer IDE. On the left, the Java Package Explorer shows a hierarchy of Java packages and classes. A specific class, `Card`, is selected, and its source code is shown in the central editor pane. The code contains several methods, including `setCard` and `setCardValue`. A large green arrow points from this code towards the center-right of the screen, where a UML sequence diagram is displayed. This diagram illustrates the execution flow and state transitions of the `setCard` and `setCardValue` methods. The diagram is annotated with text boxes:

- Integrated with the Java Package view**: Located above the Java Package Explorer.
- Select method to be visualized using UML**: Located below the Java code editor.
- Alternate abstract view**: Located above the sequence diagram.
- Leverages UML 2.0 sequence diagram constructs for loops, conditionals, etc...**: Located next to the sequence diagram.
- "Topic" diagram for method is automatically updated/refreshed when method is updated**: Located above the right-hand UML browser.

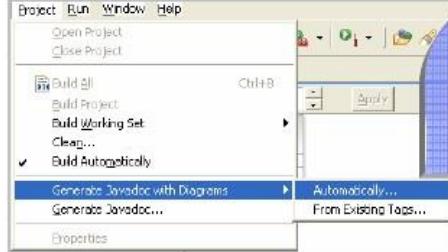
UML Enhancements:

JavaDoc with Embedded UML Diagrams

Produce enriched JavaDoc

UML diagrams right on the pages

Completely integrated with hyperlinks



All Classes
BlackjackConsole
BlackjackHand
Card
ConsoleApplet
ConsoleCanvas
ConsolePanel
Deck
Hand
TestBlackjackHand
TextIO

Package Class Use Tree Deprecated Index Help

PREV CLASS NEXT CLASS
SUMMARY NESTED FIELD CONSTR METHOD

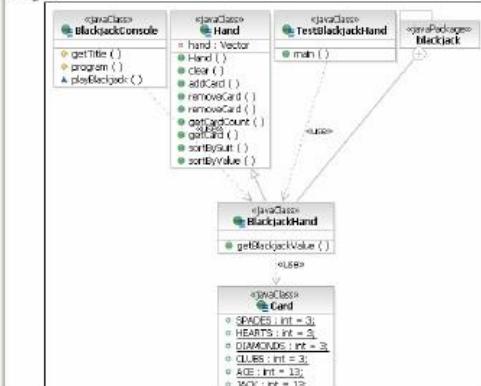
FRAMES NO FRAMES
HTML HELP CONSTR METHOD

blackjack
Class BlackjackHand

java.lang.Object
└─blackjack.Hand
 └─blackjack.BlackjackHand

public class BlackjackHand
extends Hand

Diagram:



UML Enhancements: Interaction modeling

Interactions are expressed more effectively using UML 2.0 constructs

Support specification of test scenarios

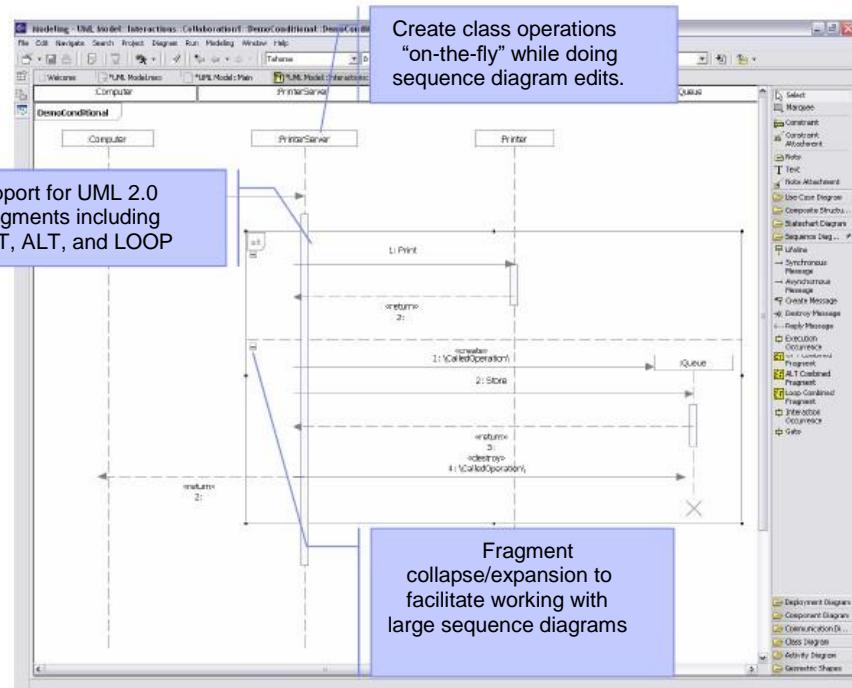
Loop, alt, opt

Interaction fragment references

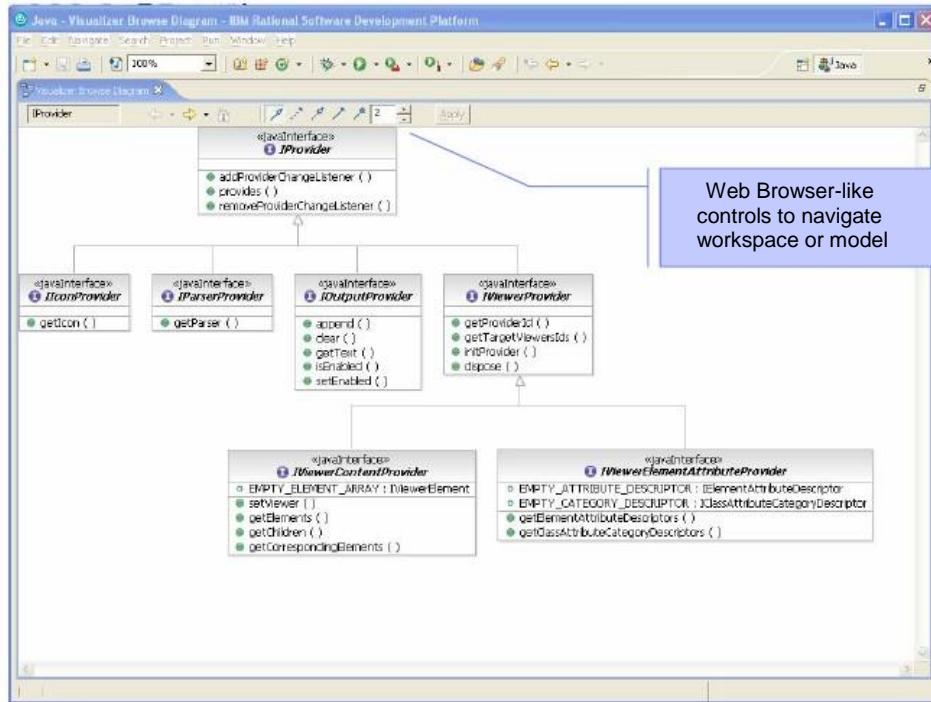
Interactions can be rendered as either sequence or communication diagrams

Sequence diagram editing improvements

Ordering and reordering



UML Enhancements: Browse Diagrams



Enables users to understand and discover models and applications without having to create or maintain diagrams

Team: RequisitePro integration

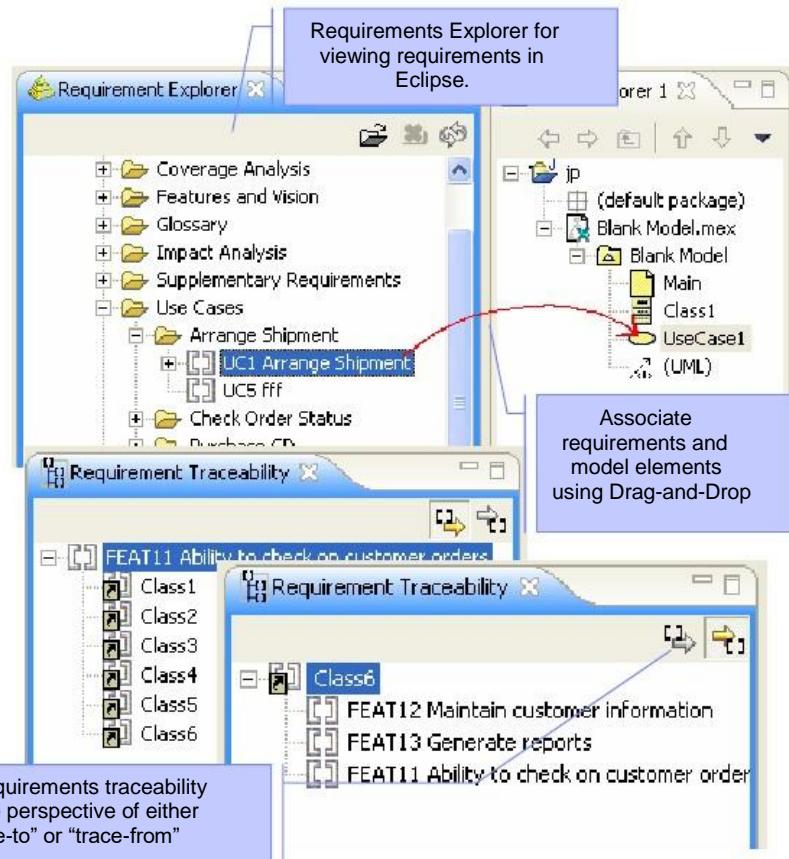
Open and browse multiple RequisitePro projects

See requirements, packages, and views

Associate requirements with model elements via drag and drop

Create model elements from requirements

Customizable synchronization



Team: Process Guidance



Ease of Use

Integration with Rational Unified Process

Tool Mentors provide guidance for activities

User customizable views with user defined content

Search is integrated with Eclipse search

Improved navigation of RUP

RUP Navigator

Role: System Analyst

The System Analyst role leads and coordinates requirements elicitation and use-case modeling by outlining the system's functionality and delimiting the system; for example, identifying what actors exist and what use cases they will require when interacting with the system.

Topics

- Description
- Related Information
- Staffing
- Further Reading

System Analyst

responsible for

RUP Advisor

Tool Mentors (2)
Tool Mentor: Working with Rational Architect
Tool Mentor: Use Case Diagrams

Artifacts (2)
Artifact: Actor
Artifact: Use-Case Model

Checkpoints: Use Case
Guidelines: Use Case
Guidelines: Activity Diagram in the Use-Case Model
Report: Use Case <use-case name>
Use Case Specification: <Use-Case Name>

Activities (2)
Activity: Review Requirements
Activity: Detail the Software Requirements

Properties RUP Advisor

Team: ClearQuest integration

The screenshot displays the IBM Rational Software Development Platform interface, specifically the ClearQuest module. The window title is "ClearQuest - IBM Rational Software Development Platform".

Left Sidebar (Navigator):

- ClearQuest
- Personal Queries
- MyTables
 - And
 - Owner = "[CURRENT_USER]"
 - State in ("Opened", "Open")
 - PathLevel in ("ClassPOS", "record_type = 'Defect'")
- Public Queries
- Aging Charts All Projects
- Classic POS Project
- Classified Web Project
- Customers
- Distribution Chart-All Projects
- Defects by Priority
- Defects by State
- Active Defects by Owner
- Active Defects by Priority
- Active Defects by Severity
- Defects by Severity, Owner
- Ellipsis State, Owner
- Email Rules
- Report Formats
- Reports
- Trend Charts All Projects
- UCNSystemQueries
- UCNSystemQueries
- All Change Requests
- All Defects
- All Defects
- All Unresolved Defects
- Defects for Specified Component

Central Area:

ClearQuest Query Results (Rev_2003_06_009KLSC)

ID	Headline	Severity	Owner	State
CLSI00000076	The System Sometimes Crashes When Ordering CDs	1-Critical	admin	Submitted
CLSI00000077	The system takes forever when the customer orders a CD	1-Critical	admin	Submitted
CLSI00000094	Registered user cannot proceed to Cashier	1-Critical	local	Submitted
CLSI00000091	StockReorder message not correct	1-Critical	local	Submitted
CLSI00000095	Wrong error for incorrect password	1-Critical	local	Submitted

Context Menu (Right-clicked on row CLSI00000095):

- Change State
- Assign...
- Modify...
- Utilities
- Close...
- Postpone...
- Details
- Expand Selected
- Refresh
- Duplicates...

Bottom Panel:

Query: Public Queries/Submitted High Severity Defects

Tasks: Problems

ClearQuest Record Details

ClearQuest Chart

ClearQuest Record Availability: 2003-06-009KLSC.mpr

Run | Notes | Resolution | Attachments | History | POC | Iterations | Test Data | Environment | Unified Change Management | ClearCase | Requirements

ID: CLSI00000092 Status: Submitted

Deadline: Wrong error for incorrect password

Sub Project: ClassicPOS Keywords:

UON Project:

Bugfix:

Severity: 1-Critical Symptoms:

Owner:

Description: If password is incorrect during login, the wrong error (invalid user ID) is displayed. The error should be "invalid password".

Total: 8 Type: Defect Selected: 1

Easy access to queries, charts, & reports

Hierarchical result set view shows parent-child relationships

Console, SQL Query & Properties views

View record forms, charts and reports

Team: ClearCase integration

ClearCase Navigator view with integrated UCM activities

The screenshot shows the ClearCase - Eclipse Platform interface. On the left, the 'ClearCase Navigator' view displays a tree structure of ClearCase streams and their contents, including 'My Activities', 'Classics', 'html', 'InvStat', 'jsdl2.1', 'lost+found', 'processes', 'Proj', 'RobotIDatabase', 'Rose', and 'Source'. Under 'Source', there are 'com' and 'natural' branches, with further sub-folders like 'admin', 'Business', 'servers', 'servers', and 'util'. A dashed line connects the 'Source' node to the 'ClearCase Details' view. On the right, the 'ClearCase Details' view shows a table of selected version information for files like AdminFrame.java, CDAdmin.java, etc. A callout box points to this view with the text: 'ClearCase Details view shows selected version information'. Below it, the 'ClearCase History' view shows a detailed history of changes for the file Logon.java, listing events such as checkouts, creates, and updates. A callout box points to this view with the text: 'Display version history, view & update config spec, display search results'.

ClearCase Details view shows selected version information

Display version history, view & update config spec, display search results

Team: Model compare & merge

Model differences & conflicts

Description of selected difference or conflict

Choose view type

Diagram view of selected difference or conflict for contributor

The screenshot shows the Rational Team Concert interface for model comparison and merging. At the top, a toolbar and menu bar are visible. Below the menu is a tab bar with tabs for 'n1.emx', 'Welcome', 'Con1.emx', 'UML Model: Main', and 'Compare (Base.emx>Con1.emx>Con2.emx)'. The main area is divided into several panes:

- Structural Differences:** A tree view showing conflicts. It lists:
 - Change and Change conflict:
 - Changed "width : EDouble" to "128.0"
 - Changed "x : EDouble" to "96.0"
 - Change and Change conflict
 - Change and Change conflict
- Merge: CompareMerge/Con1.emx:** A workspace for merging changes. It contains two classes, Class1 and Class2, each with a red border.
- Left: CompareMerge/Con1.emx:** A diagram view showing a class hierarchy with Class1 at the top, Class2 below it, and ClassCon1 as a local class.
- Ancestor Left: CompareMerge/Base.emx:** A diagram view showing the same class hierarchy as the left pane.
- Right: CompareMerge/Con2.emx:** A diagram view showing the same class hierarchy as the left pane. It includes a 'Accept' button and a 'Reject' button.
- Properties, Diagram, Explorer, EMF Properties, EMF Tree:** Tabs located at the bottom of the central workspace.

Summary: Key Features Rational Software Architect SE

Architecture Support

- Java, J2SE, C++
- UML2 Modeling
- Architecture Discovery via Application Analysis
- Patterns and Transformations



Team Environment

- Enhanced Compare / Merge
- Integrated RequisitePro Views
- Process Advisor
- CC and CQ fully integrated

Open Platform

- Based on Eclipse 3.4 Shell
- Testing and Team tools work together



Şabloane e-business

Şabloane

- Şablon
 - descrie o problemă cu repetări multiple în practică, într-un anumit context
 - descrie o soluţionare ce a fost validată prin utilizări repetitive în practică
- Reutilizare
 - clase şi obiecte multiple, între care există relaţii şi un comportament global ce pot fi reutilizate
- Şabloanele sunt folosite în implementare, dar şi pentru activităţile de analiză şi proiectare

Definiții

- Un **șablon orientat-obiect** reprezintă un model format dintr-un număr de clase ce lucrează împreună în rezolvarea unei probleme din domeniul tehnic sau economic.
- Un **șablon de proiectare** este un șablon orientat obiect ce descrie o soluție pentru rezolvarea unei probleme de proiectare.
- Un **șablon de analiză** este un șablon orientat obiect ce descrie o soluție pentru o problemă de business/analiză.

Fundamente teoretice (1)

- În dezvoltarea unei aplicații informaticice se pot întâlni situații asemănătoare celor întâlnite anterior
 - indicată reutilizarea unor soluții deja aplicate și validate în practică
- De obicei există numeroase detalii care diferențiază problemele deja rezolvate de situațiile curente, asemănarea existând doar în punctele esențiale

Fundamente teoretice (2)

- Un şablon reprezintă abstractizarea unei probleme
 - este un model de soluţionare pentru o problemă apărută într-un anumit context, verificat în practică
- Un şablon este descris prin:
 - **nume**
 - fiecare este identificat printr-un nume unic
 - **scop**
 - problema pe care trebuie să o rezolve un şablon
 - **soluţionarea**
 - modul de soluţionare propus de şablon pentru problemă
 - **participanţi şi colaboratori**
 - entităţile implicate în şablon
 - **consecinţe şi implementare**

Fundamente teoretice (3)

- Un şablon trebuie să se refere la o problemă întâlnită în mod repetat
- Soluţionarea trebuie să fie o rezolvare efectivă a problemei şi nu numai formulări de principii sau consideraţii teoretice
- Aplicarea unui şablon trebuie făcută numai în contextul pentru care a fost definit
- Şabloanele pot fi combinate în diverse moduri în vederea soluţionării de probleme complexe

Motivele utilizării şabloanelor

- Reutilizarea soluțiilor
- Stabilirea unei terminologii comune
- O perspectivă de nivel înalt (de abstractizare) a problemei cât și a procesului de proiectare/analiză
- Prin reutilizarea soluțiilor se beneficiază și se învață din experiența altora
- Nu mai este necesară reinventarea de noi soluții pentru rezolvarea unor situații comune deja întâlnite în practică
- Comunicarea și lucrul în echipă necesită un vocabular și un punct de vedere comun asupra problemei de rezolvat, lucru permis de şabloanele de proiectare/analiză ce furnizează un punct de vedere comun în timpul etapelor de analiză și proiectare a unui soft

Avantaje ale şabloanelor (1)

- **Îmbunătăţirea comunicării în echipă şi a studiului individual**
 - Acest lucru se datorează membrilor tineri ai echipei, motivaţi de cei experimentaţi, să înveţe lucruri noi şi să aplice instrumentele puternice cunoscute de aceştia.
- **Îmbunătăţirea posibilităţii de modificare a codului program**
 - Motivul este faptul că întotdeauna există factorul timp, hotărâtor în orice proces de dezvoltare de soluţii software, iar aplicarea şabloanelor permite modificarea mai rapidă a produsului software în timp util, cu minim de efort.
- **Şabloanele ilustrează principiile de bază ale abordării orientate-obiect**
 - De obicei sunt utile în creşterea înțelegerii principiilor obiectuale de bază.

Avantaje ale şabloanelor (2)

- **Facilitarea procesului de proiectare și de analiză**
 - Oferă o alternativă față de ierarhiile laborioase de clase.
 - Aplicarea şabloanelor permite dezvoltarea de soluții complexe fără ierarhii vaste de clase.
- **Adoptarea de strategii îmbunătățite în proiectare și analiză**
 - Câteva strategii sugerate sunt: proiectarea interfețelor, practicarea compunerii față de ierarhia claselor, identificarea parametrilor ce variază în contextul problemei urmată de încapsularea acestora.
- **Îmbunătățirea productivității proceselor de dezvoltare software, creșterea consistenței între aplicații**
 - datorate reutilizării, cât și aplicării repetate.

Avantaje ale şabloanelor (3)

- Oferă mai mult decât o simplă reutilizare de cod sursă
 - datorat nivelului înalt de abstractizare al reutilizării, astfel putându-se aplica pe orice platformă sistem.
- Se pot combina în rezolvarea unor probleme mai dificile, complexe
- S-au bucurat de o largă acceptare, numărul lor a crescut exponențial
 - exprimă o realitate și furnizează suportul necesar reducerii efortului de dezvoltare de aplicații software mai mult decât orice carte, lucrare sau site Web de specialitate.

Dezavantaje ale şabloanelor

- Necesită învățarea unui număr foarte mare de şabloane
 - Implică multă muncă în cunoașterea acestora în vederea aplicării corecte.
- Sindromul NIH (not-invented-here)
 - Există neîncredere în munca altora, de fapt pentru persoanele pentru care ceea ce nu este făcut personal nu reprezintă un lucru bun.
- Nu includ cod sursă, cu excepția şabloanelor de proiectare.
- Şabloanele pot deveni un domeniu haotic/superficial
 - Cu cât se va dovedi sau înțelege mai mult valoarea acestora tot mai multe persoane vor începe exploatarea lor în vederea creșterii vânzărilor și mai puțin în vederea utilității și aplicabilității.

Obiecte (1)

- În mod tradițional obiectele conțin date și metode
- Noua perspectivă a şablonelor definește un obiect ca o entitate cu **responsabilități**
- Noua vedere a obiectelor:
 - permite focalizarea asupra a ceea ce pot face obiectele, nu numai asupra simplei lor modalități de implementare
 - permite o mai bună selecție și definire a obiectelor
 - definirea obiectelor este mai flexibilă
 - focalizarea pe ceea ce fac obiectele permite aplicarea moștenirii diferit din punct de vedere al comportamentului acestora
 - gândirea în termeni de responsabilități facilitează definirea interfeței publice a obiectelor
 - unui obiect cu responsabilități i se poate cere îmbunătățirea acestora, dar informația pentru care un obiect este responsabil nu este în interiorul acestuia, nu implică nimic din interiorul obiectului.

Obiecte (2)

- **Încapsularea** - modul de ascundere a datelor obiectelor
 - aceasta se poate aplica și pentru metode, subclase sau alte obiecte
- **Moștenirea** – exprimă relațiile de generalizare / specializare dintre clase
 - reutilizarea se obține, de obicei, prin crearea de clase, urmată de derivarea de noi clase din clasele de bază
- Din perspectiva şabloanelor moştenirea se aplică astfel: identificarea variaţiilor și încapsularea lor
 - Primul pas constă în identificarea elementelor variabile în procesul de proiectare
 - metoda total opusă celei tradiționale în care se consideră variabil ceea ce produce schimbare în proiectare față de ceea ce se poate schimba fără reproiectare
 - Urmează încapsularea acestora, pasul II
 - de fapt, multe şabloane folosesc încapsularea pentru a crea straturi de obiecte slab cuplate

Abordarea obiectuală din perspectiva şabloanelor

- Din perspectiva şabloanelor într-un proces de dezvoltare software se parcurg următorii paşi:
 1. Identificarea şabloanelor corespunzătoare domeniului problemei de rezolvat.
 2. Pentru setul iniţial de şabloane identificat se parcurg următorii paşi de analiză:
 - a. se alege şablonul care furnizează, respectiv creează, cea mai mare parte din contextul celoralte şabloane;
 - b. se aplică acest şablon la nivel conceptual global;
 - c. se identifică şi alte şabloane suplimentare ce se vor adăuga la setul iniţial de şabloane analizat;
 - d. se repetă analiza pentru noul set de şabloane identificat.
 3. Se adaugă detaliile necesare proiectării, se extind prin metode şi clase.

Alegerea şablonului (2a)

- În alegerea şablonului se analizează câte o pereche de şabloane ce se pot aplica o dată
 - Întotdeauna un şablon existent într-un sistem stabileşte relaţii cu celealte existente, prin crearea unui context pentru acestea
- În etapa de analiză se începe cu identificarea relaţiilor dintre şabloane, cât şi a modul în care se stabilesc aceste relaţii
 - Acestea conduc la identificarea contextului în care se pot aplica şabloanele, dar şi a celor ce creează context pentru alte şabloane.

Alegerea şablonului (2a)

- Un concept folosit în etapa de analiză este **şablonul seniormost** – acel şablon ce impune constrângeri asupra a ceea ce vor face celelalte şabloane.
 - Există un şablon care defineşte modul de comportament al altor şabloane?
 - Există două şabloane care se influenţează reciproc?
 - Se poate defini un şablon fără altul, există un şablon ce are nevoie de un altul?

Metoda de realizare matriceală a sistemelor informaticice din perspectiva şabloanelor

- Analysis Matrix - metodă de analiză a unei aplicații e-business
 - permite identificarea, coordonarea și regăsirea elementelor variabile în concepe obiectuale din perspectiva şabloanelor
- Pașii necesari acestei metode sunt:
 - Identificarea celor mai importante caracteristici, cerințe, pentru un caz de utilizare și organizarea acestora într-o matrice bidimensională.
 - Procesarea celorlalte cazuri de utilizare urmată de extinderea matricei, dacă este cazul. Fiecare caz se procesează independent de celelalte.
 - Extinderea metodei matriceale cu noi concepte.
 - Liniile matricei identifică regulile problemei.
 - Coloanele identifică cazurile speciale.
 - Identificarea şabloanelor.
 - Dezvoltarea unui model global pentru sistemul informatic.

Observații (1)

- Construind matricea bidimensională pentru celealte cazuri din problemă (respectiv cazul 2, apoi cazul 3, ...) se identifică lipsurile și inconsistența în informații
- Beneficiarii aplicațiilor software ridică cel mai adesea probleme.
 - Ei își cunosc cel mai bine problema de rezolvat dar din păcate nu gândesc la nivel conceptual ci pe cazuri particulare, folosesc termenul de întotdeauna drept similar al termenului de obicei, sau niciodată față de rareori.
- Toate aceste lucruri duc la acele inconsistențe ce trebuie identificate încă din etapa de analiză.

Observații (2)

- Fiecare linie a matricei bidimensionale reprezintă un mod specific de implementare a conceptelor generale identificate din analiză.
- Fiecare coloană reprezintă un mod specific de implementare a cazurilor de utilizare identificate.
- Pentru fiecare linie sau coloană se identifică un şablon sau o clasă de obiecte, apoi se elaborează un model de proiectare a sistemului de realizat la nivel global, conceptual.

Observații (3)

- Aplicabilitatea acestei metode se justifică în special, în cazul problemelor complexe, cu multe cazuri specifice.
- Se analizează fiecare caz în parte, se identifică caracteristicile și comportamentul comun tuturor cazurilor, după care se construiește matricea de analiză care permite implementarea soluției software prin şabloane.
- Această metodă permite capturarea tuturor aspectelor particulare ale unei probleme.

Şabloane pentru aplicaţii de comerç electronic tip B2B

- Şabloanele de analiză (business pattern)
 - au un caracter conceptual şi sunt utilizate pentru a crea modele adecvate şi flexibile care să descrie comportamentul proceselor de afaceri studiate.
- Şabloanele de proiectare
 - propun structuri şi relaţii care asigură rezolvarea, în contexte bine definite, a unor probleme de proiectare.

Şabloane de analiză

- **Şabloanele de analiză** descriu o soluție pentru probleme uzuale ce aparțin domeniului de afaceri/analiză a unei aplicații
 - se caracterizează prin mai multă specificitate decât cele de proiectare, descriind o parte din domeniul de afaceri studiat
- În această categorie se includ următoarele şabloane: Item-Item Description, Business Entity, Contact Point, Place și Shipping/Billing.

Exemple de şabloane de analiză

- **Item-Item Description**

- Este unul dintre cele mai uzuale şi utile şabloane de analiză fiind descris drept colecţia de obiecte ce împarte aceeaşi descriere dar cu instanţe diferite.

- **Business Entity**

- Uzual, orice organizaţie dezvoltă relaţii cu alţi parteneri de afaceri, persoane fizice sau juridice. Un şablon tip Business Entity descrie relaţiile dintre diferite tipuri de organizaţii şi partenerii săi de afaceri.

Exemple de şabloane de analiză

- Contact Point

- Şablonul tip Contact Point descrie modul de interacţiune organizaţie-parteneri de afaceri.
- Subclasele acestui şablon trebuie să descrie cel puţin două lucruri: modul de transmitere de informaţii de la organizaţie la parteneri, cât şi modul de catalogare al acestora (label info).

- Shipping/Billing

- În general, în orice aplicaţie de comerç electronic tip B2B este necesară livrarea către client a produselor/serviciilor sau informaţiilor, cât şi facturilor întocmite, atât fizic cât şi electronic.
- Şablonul Shipping/Billing descrie tocmai modul de transmitere al acestora către client.

Exemple de şabloane de analiză

- **Place**

- Majoritatea organizaţiilor implicate într-o afacere online au nevoie de urmărirea zonelor în care s-au dezvoltat procesele de afaceri.
- În cazul firmelor internaţionale se impune şi cunoaşterea orelor, perioada vacanţelor, legile privind taxele, etc.
- Şablonul Place descrie modul de urmărire a tuturor acestor informaţii utile.

Şabloane de proiectare

- **Şabloanele de proiectare** descriu o soluţie pentru problemele uzuale ce apar în etapa de proiectare a sistemelor informatiche
- Cele mai întâlnite sunt Singleton, Proxy şi State
- „Gang of Four” (GoF) - Design Patterns:
Elements of Reusable Object-Oriented Software,
Gamma, Helm, Johnson, and Vlissides.
- Ian Shalloway, James R. Trott, Pattern Oriented
Design:Using Design Patterns From Analysis to
Implementation, Addison-Wesley, 2000

Exemple de şabloane de proiectare

- Singleton

- Acest şablon este de tip creaţional şi asigură o singură instanţiere, în orice moment, pentru o clasă.
- Şablonul Singleton poate fi aplicat în cazul profilelor clienţilor unde este necesară menţinerea informaţiilor de bază ale acestora (nume, număr de telefon, etc.).

Exemple de şabloane de proiectare

- Proxy
 - Şablonul Proxy indică modul de reprezentare a unui obiect ce nu este momentan în memorie.
- Să presupunem un ecran de căutare de clienți în care utilizatorul poate defini un criteriu de căutare și obține lista clientilor ce îndeplinesc respectivul criteriu.
 - Există vreo posibilitate de minimizare a volumului informațiilor trimise prin rețea?
 - Toate aceste se pot realiza printr-un şablon proxy. Aplicând acest şablon se reduce volumul informațiilor transmise prin rețea pentru că inițial se aduce doar câtă informație este necesară identificării fiecărui obiect client (nume), iar în momentul alegerii din sursă a unui client, obiectul ClientProxy aduce obiectul client din baza de date.

Exemple de şabloane de proiectare

- State

- Şablonul State descrie modul de implementare al unui obiect ce-şi poate schimba comportamentul la schimbarea stării lui interne.
- Ideea este de a construi un obiect ca o colecție de mai multe obiecte stări, ce prezintă comportamente diferite cerute de obiect pentru fiecare stare în care poate fi.
- Fiecare comportament ce este afectat de starea internă a obiectului este implementat în obiectele tip stare ca metodă.

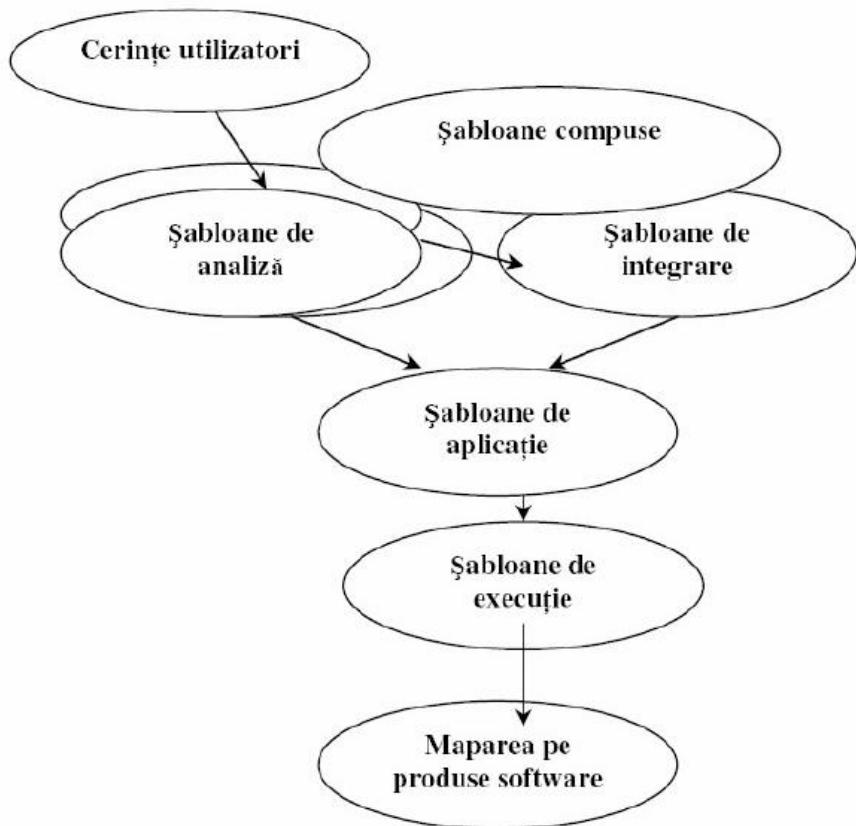
Aplicarea şabloanelor în soluţii B2B

- Rolul dezvoltatorilor IT este de a evalua problema de afaceri şi de a construi soluţii software care să o rezolve
 - Necesită timp
 - Soluţia: utilizarea şi aplicarea experienţei arhitecţilor IT din domeniu
- Pentru a putea captura experienţa dobândită de alţi experţi IT trebuie creat un depozit de valori, de soluţii deja dezvoltate în practică, pe baza cărora se pot dezvolta soluţii noi
- Această tehnică a reutilizării implică câştig de timp, bani şi efort iar într-un proces de dezvoltare soluţii IT, facilitează obţinerea unei soluţii solide, funcţionabile.

Şabloanele e-business IBM

- Aceste şabloane permit implementarea cu succes a soluţiilor B2B aplicând principiul reutilizării componentelor
- Abordarea din perspectiva şabloanelor se bazează pe un model structurat pe straturi de elemente ce pot fi exploataate şi aplicate în orice metodologie de realizare de sisteme informatiche
 - Fiecare strat este astfel structurat încât detaliile fiecărui se construiesc pe baza stratului anterior.

Modelul structurat de şabloane pentru aplicaţii e-business



Şabloanele e-business IBM

- Modelul include:
 - **şabloane de analiză** (business patterns) - identifică interacțiile dintre utilizatori, date și reguli de afaceri
 - **şabloane de integrare** (integration patterns) - realizează legătura dintre şabloanele de afaceri atunci când o soluție B2B nu poate fi descrisă printr-un singur şablon de afaceri, ajută la combinarea şabloanelor business pentru aplicații complexe
 - **şabloane compuse** (composite patterns) - reprezintă combinația dintre primele două straturi de şabloane
 - **şabloane de aplicație** (application patterns) - descrie, la un nivel conceptual, interacțiile dintre aplicații (reguli de afaceri) și date din cadrul primelor două categorii de şabloane

Şabloanele e-business IBM

- Modelul include:

- **şabloane de execuţie** (runtime patterns) - descrie structura logică, de middleware, suportată de şabloanele tip aplicaţie, înfăţişând nodurile majore, rolurile şi interfeţele dintre acestea
- **implementări software** testate din punct de vedere practic (product mappings) - identifică implementările software testate pentru fiecare şablon tip runtime (execuţie)
- **linii directoare** (best-practice guidelines) ce ghidează proiectarea, dezvoltarea, cerinţele non-funcţionale, realizarea şi managementul aplicaţiilor dezvoltate.

Şabloane de analiză

- Descrie relațiile dintre utilizatori și organizatie, regulile de afaceri cât și datele ce vor fi accesate.
- Există patru tipuri de şabloane de afaceri de bază:
 - **self-service** (user-to-business): utilizatorii interacționează direct cu organizația prin Internet sau Intranet
 - Ex: aplicații Web simple
 - **information aggregation** (user-to-date): utilizatorii pot extrage volume mari de informație
 - Ex: managementul cunoștințelor, inteligența afacerilor
 - **collaboration** (user-to-user): Internetul suportă colaborarea dintre utilizatori
 - Ex: email, chat, video conferințe
 - **extended enterprise** (business-to-business): aplicații ce fac legătura între multiplele procese de afaceri între organizații diferite
 - Ex: EDI, SCM.

Şabloane de integrare

- Permit combinarea mai multor şabloane de afaceri în vederea rezolvării unei probleme.
- Principalele şabloane de acest tip sunt:
- **access integration:** permit integrarea unui număr de servicii printr-un punct comun de acces
 - Ex: portaluri
- **application integration:** permite integrarea unui număr de aplicații și surse de date fără cererea directă a utilizatorilor, cum ar fi fluxuri manageriale, brokeri de mesaje.

Custom Design

- řabloanele de afaceri și de integrare pot fi combinate pentru implementarea unei soluții e-business specifică, cum ar fi Custom Design



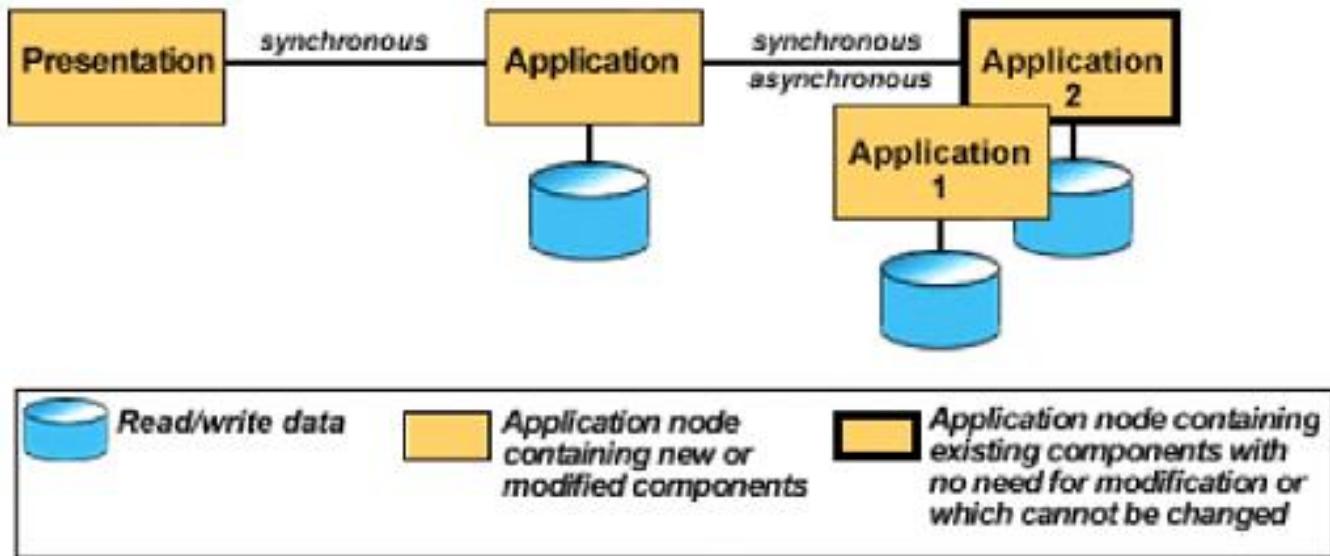
Şabloane de aplicație

- După identificarea şabloanelor business, se definesc componentele logice la nivel global și modul lor de interacțiune, ceea ce reprezintă un **şablon de aplicație**.
- De obicei un şablon de analiză are multiple şabloane de aplicație.
 - poate conține componentele logice ce descriu stratul de prezentare al aplicației (presentation tier), stratul regulilor de afaceri sau stratul de backend al aplicației.

Şabloane de aplicaţie

- Şablonul de aplicaţie împarte aplicaţia într-un număr de componente conceptuale de bază, identificând obiectivele acesteia.
- Ex. şablonul business – Self-Service
 - scopul este de a permite utilizatorilor accesul la informaţiile din organizaţie de tip back-end.
 - în acest şablon componentele conceptuale sunt: aplicaţia de prezentare, aplicaţia ce descrie regulile de afaceri şi aplicaţiile existente în firmă (back-end) ce reprezintă stratul de persistenţă a datelor.

Self-Service::Directly Integrated Single Channel

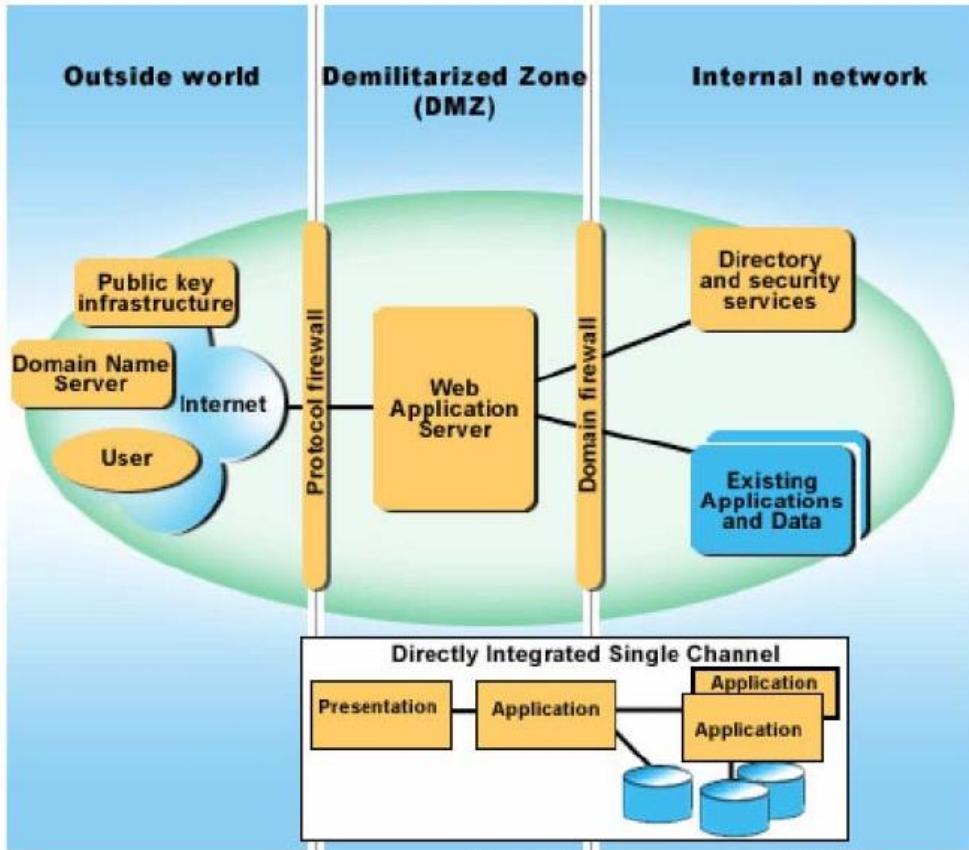


Şabloane de execuţie

- În şabloanele de execuţie (runtime), atenţia se focalizează pe nodurile logice şi pe locul de plasare în întreaga structură a reţelei
 - Şabloanele aplicaţie se dezvoltă prin adăugarea de noi funcţii explicite, fiecare fiind asociată unui nod de execuţie ce pot exista pe maşini diferite sau pe aceeaşi maşină
- Continuând exemplul anterior, se trece la identificarea şablonului de execuţie potrivit pentru şablonul business şi cel de aplicaţie ales

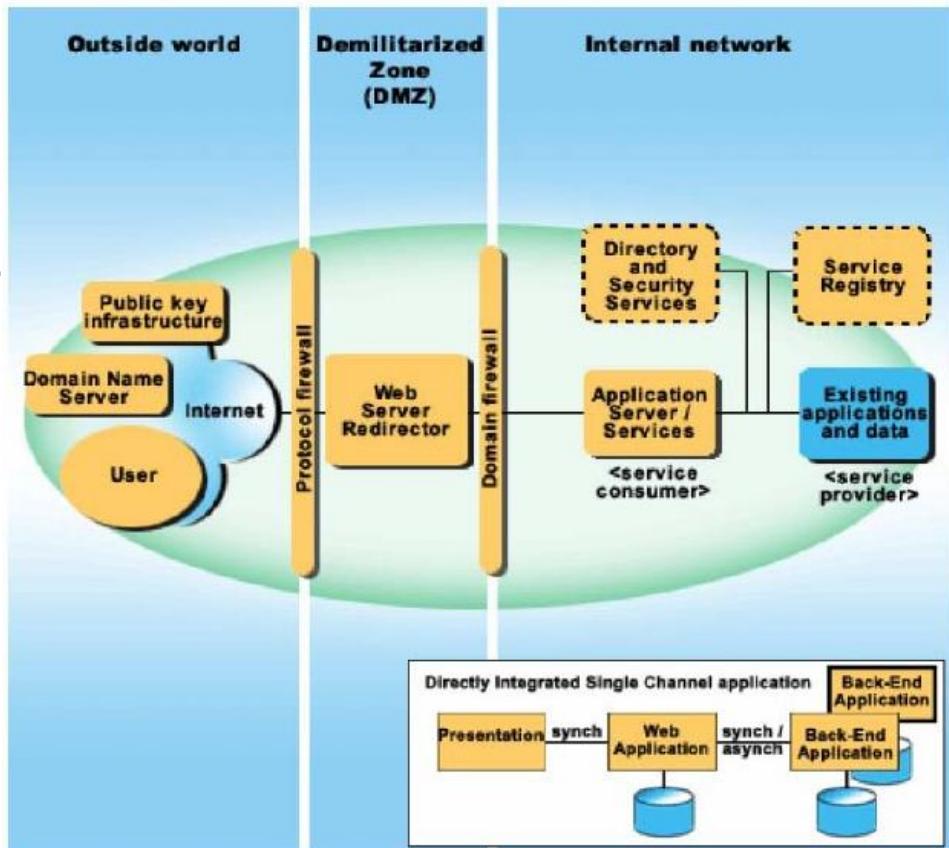
Şabloane de execuţie

- Presupunând accesul utilizatorilor la datele organizaţiei prin Internet, se impun şi anumite măsuri de securitate.



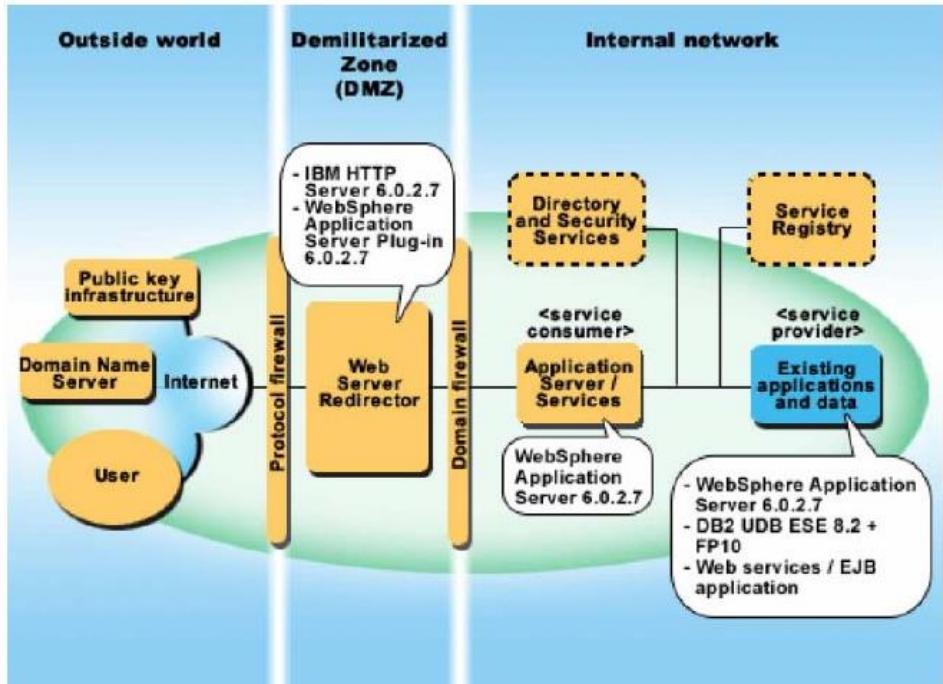
Şabloane de execuţie

- O caracteristică a şablonului de execuţie este plasarea serverului Web de aplicaţie între două firewall-uri.
- În cazul cererilor crescute de securitate există un şablon de execuţie ce împarte serverul Web de aplicaţie în două noduri funcţionale, separând funcţiile serverului HTTP de serverul de aplicaţie



Implementări software testate

- Ultimul pas în definirea structurii unei aplicații este corelarea produselor IT existente cu unul sau mai multe noduri logice de execuție.
- Fiecare mapare pe produse IT este orientată către o platformă software particulară funcție de preferințele beneficiarilor.



Sumar

- Şabloanele facilitează înțelegerea și analiza problemelor complexe, cât și divizarea lor în componente mai mici, mai ușor de administrat și de implementat.
- Prezentarea este doar o abordare succintă a tipurilor de şabloane ce pot fi aplicate în domeniul e-business.
- Teoria poate fi studiată mai pe larg pe site-ul IBM:
<http://ibm.com/developerworks/patterns/>

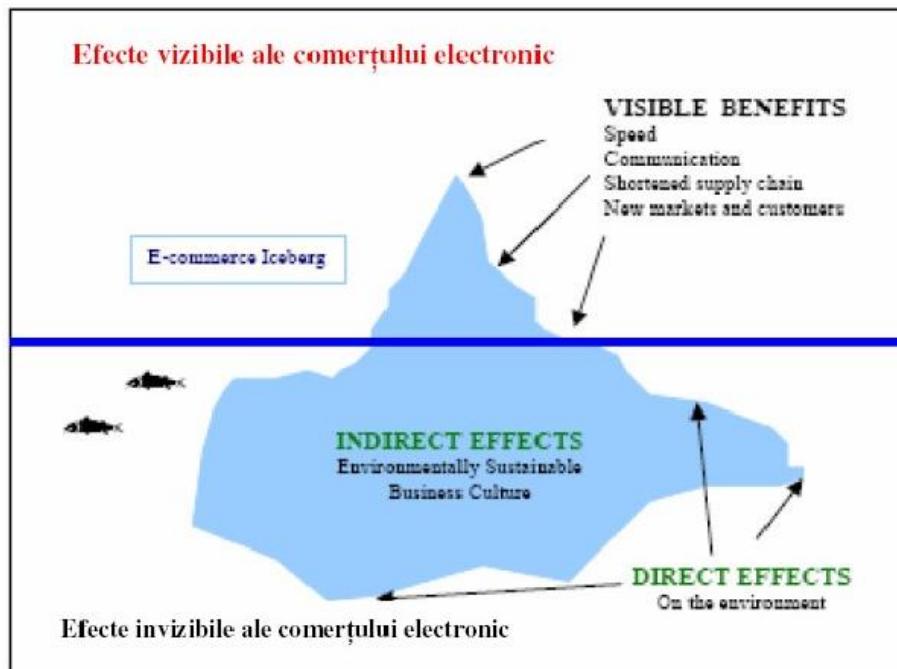
Metodologie de realizare a sistemeelor de comerț electronic tip B2B

Introducere

- Activitatea unei firme este influențată de mediul înconjurător și de factorii determinanți ai acestuia
- Mediul ambiant este deosebit de dinamic
 - Schimbări mai ales sub impactul revoluției științifico-tehnice
- Cantitatea de informații privind Internetul și comerțul electronic este mai mult decât suficientă pentru a crea confuzii
- Internetul și tehnologiile digitale ce facilitează adoptarea comerțului electronic atrag orice operatiune sau activitate la limita fizicului cu virtualul:
de la infrastructura pieței financiare și până la comunicațiile digitale

Efectele comerțului electronic (1)

- Cercetătorii aseamănă pătrunderea în domeniul comerțului electronic ca și schiatul pe un iceberg.



Efectele comerțului electronic (2)

- Partea vizibilă reprezintă beneficiile datorate comerțului electronic:
 - creșterea vitezei de derulare a afacerilor,
 - îmbunătățirea conectivității cu partenerii de afaceri și clienți,
 - globalizarea piețelor,
 - scurtarea lanțului de aprovisionare,
 - transparența afacerilor
- Ceea ce este mai interesant sunt efectele mai puțin vizibile ale comerțului electronic
- Efectele invizibile pot fi atât pozitive cât și negative asupra mediului de afaceri al firmei
- Centrul icebergului este analog culturii organizaționale și atitudinii managerilor firmei privind afacerile tip e-business în general

Premisele strategice

- Firmele trebuie să decidă **CUM** și **CÂND** să implementeze o aplicație de comerț electronic
- Alegerea trebuie să se bazeze pe o evaluare completă de costuri și beneficii, pe obiective riguroșe definite, care depind de modalitatea de fundamentare a deciziei (fundamentare economică sau strategică)
- Premisele strategice de implementare a unei soluții e-business sunt următoarele:
 - studiul de implementare a soluției e-business în firmă,
 - procesul de planificare strategică,
 - factori determinanți,
 - tipul de decizie (strategică sau tactică/curentă),
 - sectorul de activitate,
 - tipul de firma.

Studiul de implementare

- Conform studiilor o firmă este pregată pentru domeniul e-business în momentul în care se răspunde cu „DA” la următoarele afirmații:
 - Există comunicare prin email cu clienții și furnizorii firmei?
 - Se comandă prin email produse/servicii de la alți parteneri de afaceri?
 - Se vând produse/servicii proprii prin intermediul site-ului firmei?
 - Se folosește Web-ul pentru informare privind alte produse?
 - Se furnizează informații despre produsele/serviciile proprii pe site?
 - Se folosește site-ul ca mijloc de gestionare a informațiilor despre clienți?
 - Se folosește Internetul pentru plata online?

Procesul de planificare strategică a soluției e-business

- Planificarea strategică a unei afaceri e-business trebuie înglobată în afacerea firmei, trebuie dezvoltată în relație cu obiectivele majore ale firmei
- Întrebările importante la care trebuie să se raspundă când se ia în considerare un proces de planificare strategică tip e-business sunt:
 - Care sunt obiectivele offline și strategia offline dezvoltate anterior?
 - Cum va gestiona firma impactul e-business asupra clienților săi, furnizorilor sau canalelor de distribuție?
 - Care este răspunsul competitorilor săi vizavi de adoptarea e-business a firmei?
 - Care sunt limitele legale, de infrastructură, standarde?
 - Vor crește veniturile, vor scădea costurile sau ambele, datorită adoptării soluției e-business?
 - Este pregatită firma în vederea implementării soluției e-business?

Procesul de planificare strategică a soluției e-business

- Fazele elaborării unei strategii e-business sunt:
 1. stabilirea obiectivelor online;
 2. analiza în vederea redefinirii acestor obiective;
 3. selectarea unui model e-business de implementat;
 4. identificarea proceselor de afaceri implicate de obiectivele propuse;
 5. identificarea instrumentelor e-business necesare îndeplinirii obiectivelor propuse.

Determinanții endogeni ai unei strategii e-business

- În dezvoltarea unei strategii e-business este importantă înțelegerea implicațiilor implementării proceselor online asupra firmei în general
- Se recomandă luarea în considerare a **patru** astfel de factori:
 - calitatea operațională a proceselor interne – dacă Intranetul firmei asigură o comunicare eficientă între unitățile de afaceri;
 - calitatea relațiilor manageriale – se referă la calitatea desfășurării relațiilor online ale firmei cu partenerii externi;
 - noile modele de afaceri adoptate – modul în care s-a realizat integrarea proceselor online cu cele offline;
 - avantajele desfășurării proceselor de afaceri online.
- Adevarăta artă - alegerea factorului optim pentru firma, identificarea obiectivelor particulare aferente acestuia și configurarea celor mai potrivite instrumente și procese împreună cu cele offline în vederea realizării obiectivelor propuse

Decizie strategică sau tactică?

- Hotărârea implementării unei aplicații e-business se poate face pe baza unei decizii strategice sau tactice.
- Decizia strategică
 - include mare parte din aspectele unei analize SWOT
 - nu poate fi cuantificată în indicatori economici
- Decizia tactică
 - focalizată pe costuri și fluxuri de venituri
 - poate fi cuantificată în indicatori economici
- În timpul procesului strategic se poate uita tipul de decizie ales și se încearcă o cuantificare a succesului pe baza parametrilor greșită
 - de exemplu, pentru o decizie strategică se masoară rezultatele obținute prin indicatori de costuri

Este sectorul de activitate unde activează firma pregătit pentru e-business?

- Înaintea deciziei strategice privind tipul de instrumente e-business de folosit, firma trebuie să cunoască **CUM** și **CÂND** au mai fost folosite acestea.
- Este important de considerat aplicarea e-business-ului în întreg lanțul de valori, de la furnizor până la clienți.
- Extinderea e-business-ului în cadrul unui sector de activitate este influențată de:
 - caracteristicile tehnologice potrivite,
 - resursele și competențele firmelor din sectorul de activitate,
 - reglementări legale,
 - standardele privind produsele/serviciile sectorului de activitate.

Este pregată firma pentru implementarea soluției e-business?

- Instrumentele e-business (email, site Web, marketing și vânzari online, EDI, etc.) sunt în continua extindere dar fiecare firmă trebuie să găsească combinația optimă a mixului online și offline.
- Comerțul electronic este mai mult o strategie de afaceri decât o simplă tehnologie
 - Implică decizii strategice ce necesită înțelegerea, angajarea și participarea activă atât a managerilor firmei cât și a proprietarilor
- Provocarea constă în determinarea mixului optim de tehnologii ce se pot aplica pentru realizarea obiectivelor strategice.
- Toate firmele trebuie să înceapă prin înțelegerea impactului strategic al comerțului electronic asupra propriei afaceri.
- Înaintea derulării unei afaceri online un rol important revine etapei de fundamentare a strategiei de implementare a comerțului electronic.

Este pregată firma pentru implementarea soluției e-business?

Există o listă de întrebări în vederea înțelegerei impactului comerțului electronic asupra afacerii:

- Cât de rapid se dezvoltă comerțul electronic în sectorul de activitate?
- Cât de repede trebuie derulată intenția de implementare a comerțului electronic de către firmă astfel încât să câștige avantaj competitiv?
- Câți parteneri de afaceri deja au implementat astfel de aplicații?
- Care sunt factorii stimulatori ai trendului în comerțul electronic?
- Ce tehnologii sunt utilizate la implementarea comerțului electronic?
- Ce tehnologii folosesc partenerii de afaceri cât, mai ales, concurența?
- Cum influențează comerțul electronic comportamentul clientilor, furnizorilor, angajaților firmei?
- Este firma capabilă să ofere produse/servicii competitive?

eStrategy (1)

- Termenul de eStrategy folosit în diverse contexte
 - toată lumea îl folosește
- Dificil să se dea o definiție precisă a eStrategy,
dar:
- eStrategy reprezintă "puntea" dintre strategia
unei afaceri și soluțiile tehnologice alese pentru
punerea ei în practică.
 - Pentru ca o afacere să obțina valoare prin aplicarea
tehnologiei, elementul critic este să se înțeleagă modul
în care aceasta poate fi folosită pentru a crea valoare
și pentru a susține strategia de dezvoltare a afacerii
respective;
- eStrategy integrează o afacere și un context
tehnologic cu o viziune a viitorului.

eStrategy (2)

- eStrategy reprezintă strategia "din spatele" e-business.
- O alta tratare uzuală - percepere strictă de abordare centrată pe client.
 - mulți experti susțin că transformarea unei afaceri într-o centrata pe client constituie primul pas în implementarea unei eStrategy.
- “cheia problemei nu o reprezintă doar web-ul - ci
întreprinderea în sine. În timp ce abilitatea de a interacționa
și dialoga cu clientul constituie doar faza de început,
formularea unei strategii de integrare a aplicațiilor interne
(back-end) cu soluțiile de afaceri-electronice constituie cheia
unui CRM (Customer Relationship Management) de
succes”, Jeff Cladwell (consultant în solutii tehnologice
pentru dezvoltarea organizațiilor),
- În ultimă instanță, eStrategy reprezintă un plan cuprinzător,
strâns legat de esența strategiilor de marketing și de scopul
afacerii în sine, ținând totodată seama și de tehnologie.

eStrategy (3)

- eStrategy acoperă următoarele aspecte ale e-business:
 - Comerț electronic:
 - cu câtva timp în urma, acesta se referea numai la efectuarea de tranzactii electronice on-line cu clientul; în prezent prin Comerț electronic se înțelege modul în care tranzactioneaza furnizorii, partenerii si clientii;
 - Business Intelligence:
 - utilizarea informatiilor referitoare la clienti pentru a lua decizii profitabile, a afla cine sunt si a comunica mai eficace;
 - eCustomer service:
 - servirea mai buna a clientilor, 24 de ore, 7 zile pe saptamâna, câștigarea de noi clienti, consolidarea relatiilor cu clientii existenti;
 - Supply Chain Management:
 - modernizarea proceselor end-to-end pentru obtinerea unui avantaj competitiv;
 - Enterprise Business Intelligence:
 - partajarea, comunicarea si analiza mai eficienta a datelor despre clienti.

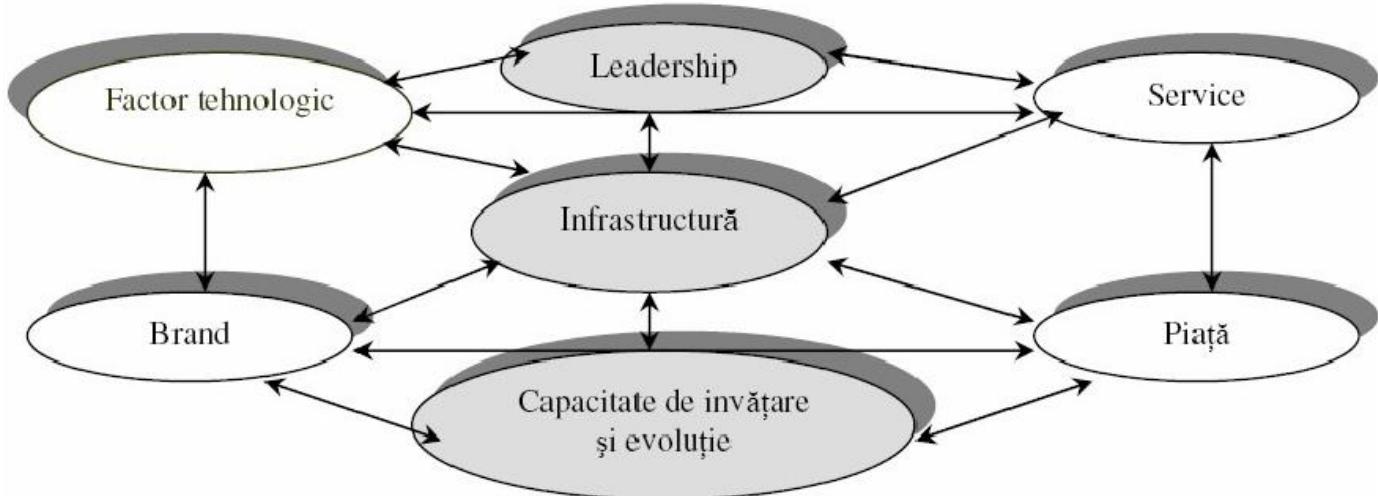
Factori strategici (1)

- Formularea unei strategii de implementare de comerț electronic este un **proces complex**, care capătă aspecte specifice funcție de mai multe elemente:
- natura operațiunilor de comerț electronic în cadrul firmei
 - Firme ce își mută parțial operațiunile pe Internet sau firme „născute” pe Internet;
 - natura ofertei firmei – producție, servicii sau mix;
 - modelul online adoptat – B2B, B2C, etc.

Factori strategici (2)

- Experiența marilor firme care au reușit arată că pentru elaborarea unei strategii de succes s-a conturat un model care ia în considerare mai mulți factori strategici:
 - factorul tehnologic,
 - piața,
 - brand-ul,
 - service-ul,
 - leadership-ul,
 - infrastructura
 - capacitatea organizației de a învăța și evoluă

Factori strategici (3)



Modelul factorilor strategici

- Modelul anterior este oarecum general și aplicabil firmelor din sectoarele industrial și servicii.
- Firmele trebuie să urmărească permanent acești factori strategici și să-și adapteze strategiile pentru a răspunde schimbărilor mediului în care operează.
- Modelul
 - are rolul de a semnala factorilor decizionali și executivi elementele strategice care trebuie avute permanent în vedere
 - este simultan și suficient de flexibil pentru a fi util firmelor mari în procesul de mutare a unor operațiuni comerciale pe Internet, dar și celor de mici dimensiuni care pornesc de la ideea exploatarii unei nișe din mediul comerțului electronic.

Leadership-ul

- Studiile de caz menționate în literatura de specialitate arată că întotdeauna în spatele succesului unor acțiuni care vizează introducerea operațiunilor de comerț electronic în ansamblul operațiunilor unei firme s-au aflat manageri de top, membri remarcabili ai staff-ului unor companii.
 - Louis Gerstner a restructurat și transformat IBM pe baza noului concept de e-business.
 - Jacques Nasser a integrat e-business în strategia globală a companiei Ford.
- De multe ori companii ai căror manageri au ignorat fenomenul comerțului electronic sau au acționat întârziat și pe baze insuficient pregătite au cunoscut eșecuri care s-au materializat în pierderi la nivelul firmelor
 - cazul firmei Levi's – care deși a intrat de Internet – datorită lipsei unei strategii eficiente a fost nevoită să părăsească temporar acest segment.

Leadership-ul

- Se constată că pentru managerii de top ai companiilor se impun unele cerințe din acest punct de vedere:
 - să păstreze un spirit vizionar deschis cu accent pe identificarea și evaluarea potențialului noilor tehnologii;
 - să nu respingă tehnologiile noi și experimentale care apar pentru că printre acestea se vor regăsi vectorii orizontului tehnologic viitor;
 - să încurajeze initiațiva și ideile inovatoare care provin din interiorul sau exteriorul firmei, identificând posibilele oportunități pentru propria companie;
 - să fie pregătiți să conceapă și să implementeze schimbări importante în strategia firmei când acest lucru este impus de apariția unui salt tehnologic.

Infrastructura

- În momentul în care necesitatea de a dezvolta activități de comerț electronic a fost identificată pentru o firmă, una dintre cele mai importante sarcini este infrastructura.
 - aspect esențial pentru succesul unei strategii de comerț electronic
- Infrastructura care privește operațiunile de comerț electronic poate fi considerată la mai multe niveluri:
 - la nivel strategic;
 - la nivel organizational;
 - la nivel fizic.

Infrastructura

- La **nivel strategic**, infrastructura trebuie concepută în acord cu impactul ce se estimează că îl va avea dezvoltarea tehnologică asupra pieței și firmelor.
 - Scopul urmărit va fi să se armonizeze concepția asupra infrastructurii cu aceste preocupări tehnologice viitoare.
- La **nivel organizational**, se urmărește modelarea unei infrastructuri unde procesele de muncă, fluxurile, structurile să fie adaptate pentru a îndeplini în mod eficace și eficient obiectivele firmei.
 - Execuția efectivă a activităților de comerț electronic se realizează prin intermediul nivelului fizic al infrastructurii, echipamentelor hardware, software-ului utilizat, precum și conexiunilor de telecomunicații.

Infrastructura

- Se impune crearea unei infrastructuri capabile să absoarbă şocurile determinate de evoluţiile tehnologice viitoare.
- Crearea unei infrastructuri funcţionale şi adaptată cerinţelor activităţilor de comerţ electronic necesită un nivel ridicat de deschidere şi comunicare între toate compartimentele şi nivelurile ierarhice ale firmei.
- Soluţia tehnică aleasă va trebui să fie scalabilă, sigură şi robustă.
- Managerii de top vor trebui tot mai mult să posedă cunoştinţe legate de tehnologiile prezente şi de perspectivele viitoare, pentru a direcţiona strategic corect dezvoltarea viitoare a firmei.

Capacitatea firmei de a învăța și de a se adapta

- Capacitatea unei firme de a reacționa, înțelege și de a se adapta la specificul operațiunilor de comerț electronic - legată direct de capacitatea organizației de a învăța.
- Crearea unui mediu care să stimuleze și să încurajeze acest proces de instruire și învățare la nivelul organizațiilor trebuie să fie o permanentă preocupare a managementului firmei.
- Procesul de învățare la nivel organizațional trebuie să fie direcționat, focalizat spre obiectivele strategice ale firmei, cuprinzând toate domeniile de interes: tehnologie, service, brand, poziționare pe piață.

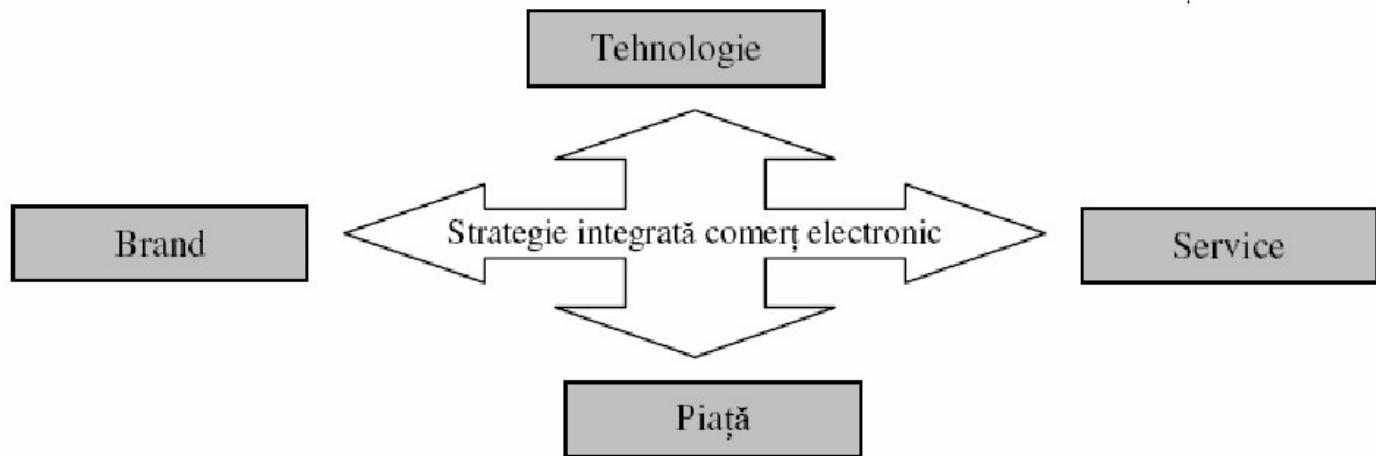
Procesul de învățare/adaptare

- Procesul de învățare
 - premisă pentru o adaptare continuă la schimbările de mediu
 - se bazează de cele mai multe ori pe feedback-ul receptat de firmă la activitățile sale.
- În cazul comerțului electronic
 - diferențierea pozitivă față de ceilalți actori de pe piață este importantă
 - urmărirea reacției clientilor și adaptarea modului de acțiune la aceste reacții sunt esențiale.
 - cele mai multe site-uri de comerț electronic dispun de mecanisme directe sau indirecte de culegere a feedback-ului de la clienți - informații sintetizate de personal dedicat și prezentate managerului firmei

Procesul de învățare/adaptare

- În acest proces continuu de receptare de feedback și adaptare sunt două aspecte:
 - Este necesară implementarea unui sistem fiabil de cuantificare, selecție și măsurare a informației de feedback.
 - Aspectele cantitative ale acestui proces sunt la fel de importante ca și cele calitative.
 - Este esențială viteza de reacție a firmei la cerințele în continuă schimbare ale mediului în care operează.
 - Capacitatea de sinteză dinamică rapidă a informațiilor de feedback poate determina o curbă de instruire și adaptare rapidă.
 - Existența unui compartiment intern dedicat analizei și dezvoltării activităților online este o investiție strategică pentru firmele care desfășoară activități de comerț electronic.

Integrarea celor patru factori strategici principali



Factorul tehnologic

- Succesul strategiilor privind activități de comerț electronic este determinat în bună măsura de atenția acordată **factorului tehnologic**.
 - Printre marile firme americane fidele permanent acestui concept se pot nominaliza SUN Microsystems, Motorola, etc.
- Factorul tehnologic comportă abordări în bună măsura diferite, funcție de momentul în care o firmă decide integrarea activităților de comerț electronic și elaborarea unei strategii în acest sens.

Firmele nou apărute

- Pentru **firmele nou apărute** pentru exploatarea acestei nișe a activităților comerciale pe Internet obiectivul tehnologic trebuie corelat cu piața și sectorul pe care va activa
- O organizație nouă îl va defini în strategia sa și va viza permanent un nivel tehnologic care să-i confere statutul de lider în domeniu sau va apela la sisteme verificate deja în practică, mai stabile, necesitând investiții mai reduse, aflându-se în urmărirea liderilor de pe piață.
- Esențial este ca tehnologia folosită de firmă să răspunda necesităților și așteptărilor clientilor
- Strategia acestor firme trebuie să răspundă unor întrebări specifice precum:
„poate tehnologia folosita sa creeze bariere de intrare unor alți competitori pe piata?”
„poate tehnologia folosita sa mareasca probabilitatea de a mentine captiv un nucleu cât mai mare de clienti?”
„se va putea mentine statutul de lider tehnologic pe piata?”.

Firme existente

- Pentru **firmele existente**, care decid să abordeze activități de comerț electronic, există aspecte specifice suplimentare care trebuie avute în vedere.
- Trebuie luate în calcul implicațiile tehnologice complete pentru firmă.
- Se va analiza dacă procesele interne actuale corespund sau cum trebuie modificate într-o abordare a operațiunilor legate de Internet.
- Managementul firmei va trebui să înțeleagă modul în care clienții privesc și folosesc suportul tehnologic oferit și să valorifice aceste informații pentru conceperea unei infrastructuri care flexibilă și adaptabilă la schimbările strategiei dedicate comerțului electronic.
- Deciziile privind componenta tehnologică a strategiei va viza minimizarea costurilor și maximizarea eficienței sistemului implementat.

Factorul piață (1)

- **Piața** este unul dintre determinanții esențiali ai strategiei firmelor privind comerțul electronic.
- Firmele creative au reușit de cele mai multe ori să obțină în practică o creștere majoră a cifrei de afaceri și a cotei de piață
 - au găsit modalități de adaptare rapidă la specificul mediului economic
 - au pus la punct o ofertă pornind de la înțelegerea cerințelor clientilor în mediul în care operează
- Pentru o firmă care vizează intrarea în domeniul activităților online esențial este să determine piața sa țintă și dacă aceasta este în mod realist deschisă unor noi intrări (dacă firma poate depăși eventualele bariere de intrare).
- În paralel, fundamentarea strategiei trebuie să ia în calcul segmentarea pieței și ritmul prognozat de creștere pe termen mediu și lung.

Factorul piață (2)

- Firmele care desfășoară activități comerciale în modul clasic trebuie să își fixeze cota de piață vizată în mediul online și factorii care pot face această cotă superioară celei offline.
- Analiza pieței trebuie să vizeze și
 - anticiparea mișcărilor concurenței
 - modul în care piața va fi afectată de intrarea unor alți actori semnificativi în operațiunile clasice de comert
- Statistica arată că în multe sectoare de activitate Internetul a indus o creștere generală a segmentelor respective, fapt posibil prin deschiderea de noi canale comerciale cu acest suport.
- Firmele care au ignorat sau au subestimat posibilitățile oferite de Internet au cunoscut scăderi de ansamblu a cotelor de piață.

Factorul servicii

- Orice firmă trebuie să cunoască așteptările clienților privind **serviciile** pe care anticipatează că le vor fi oferite
- Valoarea totală oferită de o organizație clienților săi cuprinde, pe lângă produsul de bază și această gamă de servicii.
- Există o relație directă de cele mai multe ori între gradul de atragere a clienților și nivelul serviciilor oferite.

Factorul servicii

- Serviciile aditionale reprezintă **un plus de valoare** oferit de firmă clienților
 - nu este de regulă imediat recompensat, nu se transformă imediat într-o vânzare suplimentară
 - pe baza acestei game de servicii adiționale se consolidează relația firmă-client, se obține un câștig la nivelul informațiilor dobândite privind profilul și așteptările clientilor, se creează o categorie de clienți potențiali și se dezvoltă legăturile cu cei deja existenți.
- Gama de servicii adiționale pe care o firmă poate decide să le ofere merge de la informații generale nespecifice la servicii strâns legate de oferta de bază a firmei.

Factorul servicii

- Pe majoritatea site-urilor cu activități de comerț electronic sunt cuprinse informații privind
 - cursul valutar,
 - indicii bursieri,
 - servicii meteo, etc.
- În cazul instituțiilor unde clientul poate solicita un credit există aplicații simple prin care se poate determina accesibilitatea și nivelul de încadrare la un astfel de produs financiar.
- Furnizarea unei simple adrese de email permite firmelor să trimită clienților periodic o formă de gazetă electronică – newsletter – care poate cuprinde oferte noi, oferte speciale, actualizări ale unor oferte, totul pentru a-l determina pe client să revină pe site-ul firmei și în final să realizeze o nouă cumpărare.

Factorul brand

- O dată cu dezvoltarea activităților de comerț electronic a apărut și preocuparea studierii fenomului de **brand**.
- Internetul induce clientului o nouă conduită, diferită de cea din mediul offline, astfel că din punct de vedere al firmei apar provocări specifice la care trebuie să facă față – abilitatea de a crea un brand nou, de a întări sau repoziționa brand-ul existent.
- În definirea strategiei sale privind brand-ul o firmă trebuie să răspundă întrebări cum ar fi:
- are firma capacitatea de a crea un brand puternic în mediul electronic?
- pe ce va baza brand-ul în primul rând: pe tehnologie, pe serviciile adiționale oferite sau pe poziționarea firmei în piață?
- va schimba ceva din imaginea brand-ului offline existent deja, noul brand ce va fi creat pe Internet?

Factorul brand

- Procesul de creare a unei strategii de branding pentru comerțul electronic este în mod clar diferit pentru firmele “nou-născute” pe Internet și cele existente offline.
- Cazul cel mai cunoscut și discutat privind firmele noi se referă la **Amazon.com**, care a devenit practic nu numai cel mai mare vânzător de cărți pe Internet, ci și brand-ul cel mai cunoscut de pe Internet.
- Înființată în 1995 ca cel mai mare furnizor de cărți online
- Repede și-a diversificat activitatea, prin vânzarea de cărți, DVD-uri, computere, aparatură, etc.
- Reflexia potențialului a ceea ce este comerțul electronic
- Cel mai avansat tehnologic magazin pe Internet (zeci de milioane de clienți)
- Închiriază spațiu pe platforma tehnologică unor alte magazine online

Amazon.com

Amazon.co.uk: low prices in Electronics, Books, Music, DVDs & more

amazon wikipedia

Cull Press and Jo... News (361) Leuchars Departs BBC News Guardian Scotsman North of Sco...and Journal .Mac News (1738) Apple (132)

amazon.co.uk

Hello. Sign in to get personalised recommendations. New Customer? [Start here.](#)

Get Unlimited Free One-Day Delivery Now!

Your Account | Help

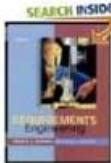
Shop All Departments

- Books
- Music, DVD & Games
- Electronics
- Computing & Office
- Home & Garden
- Toys, Children & Baby
- Jewellery & Watches
- Shoes & Accessories
- Sports & Leisure
- Health & Beauty
- DIY & Tools

Search sommerville

Similar Books to Explore

You looked at You might also consider

 Requirements Engineering: Processes... Hardcover by Gerald Kotonya, Ian... £36.05	 Software Requirements, Second Edition Paperback by Karl E. Wiegers £26.99 £18.19	 Requirements Engineering: A Good... Paperback by Ian Sommerville, Pete... £36.05
---	--	---

This phrase occurs frequently in each of these books: "traceability tables" ([learn more](#))

What Do Customers Buy After Viewing This Item?

67% buy the item you viewed	19% buy this alternative	7% buy this alternative
-----------------------------	--------------------------	-------------------------

Check This Out

Clearance Clearance Find our clearance deals across selected stores.

Autumn Deals See deals and new releases across selected stores this autumn.

DIY & Tools Visit our new store today.

Score a Hole In One! **Save** To help improve your game, we have some great offers on

Intempo IFI-01 , 2.1 Ipod Stereo...
Chuck out the stereo! Its time to spring clean your Hi-Fi! At...
[Read more](#) **£197.44**

ADVERTISEMENT

audible.co.uk® Audiobooks just got cool

Start Free Trial

Proiectarea unei aplicații de comerț electronic tip B2B

Introducere

- Abordarea tradițională a procesului de realizare de soluții software (analiza cerințelor, arhitectura sistemului, proiectare, implementare, testare și exploatare) nu mai este suficientă
- Noile linii directoare sunt:
 - **Furnizarea de servicii** (abordarea procesului de dezvoltare a sistemelor informatic din punct de vedere al serviciilor)
 - o firmă este privită ca o colecție de funcții oferite de o infrastructură IT și un set de servicii integrate în scopul îndeplinirii obiectivelor de afaceri
 - arhitectura sistemului informatic al unei firme trebuie proiectată ca o colecție de servicii ce comunică prin protocole interne.
 - **Asamblarea și reasamblarea rapidă a soluțiilor IT**
 - presupune o mare flexibilitate a soluțiilor propuse a.î. echipele de dezvoltare să fie capabile să asambleze și să integreze rapid pachete de soluții soft în scopul satisfacerii noilor cerințe de afaceri
 - se impun relații foarte strânse între analiști și arhitecți IT și o abordare managerială optimă a componentelor IT de integrat.
 - **Focalizarea pe reutilizare și management eficient**
 - extindere a tehnicii de reutilizare pe tot parcursul ciclului de viață al sistemului informatic echivalentă cu extinderea reutilizării de la biblioteci de cod la procese de afaceri, analiza cerințelor, specificații de proiectare, etc.

Aplicarea şablonelor în procesul de realizare a unei aplicaţii de comerç electronic tip B2B

faza de analiză şi proiectare

Faza de analiză a aplicației B2B

- Sistem de comerț electronic
 - mai mult decât modificarea sau optimizarea unor funcții organizaționale prin utilizarea tehnologiilor IT sau a unui personal specializat
 - nu constă doar în schimbarea structurii organizației ci reinventarea conceptelor unei organizații
- Câteva exemple de schimbări într-o organizație sunt:
 - modificarea granițelor sistemului informatic;
 - suporul fizic al sistemului devine aproape în totalitate virtual;
 - resursele utilizate sunt formate din informații;
 - automatizarea operațiilor.

Exemplu

- Cazul unui IMM care dorește implementarea unei soluții de comerț electronic B2B Sell-Side.
- Modelul se caracterizează prin:
 - un singur vânzător către mai multe ramuri;
 - deține propriul catalog de produs, gestiune de stoc, relații cu firmele cumpărătoare, gestiunea ordinelor de la clienti;
 - reperarea cumpărătorilor și întreținerea de relații de parteneriat de lungă durată;
 - prețul de vânzare al produselor este determinat prin cantitate, locația distribuirii și data de distribuire;
 - responsabilități: gestionarea datelor privind produsele achiziționate de la furnizori, gestionarea ordinelor clientilor și a stocului, integrarea cu sisteme informatiche diferite aparținând clientilor cât și sistemele proprii vânzatorului (cum ar fi ERP, SCM, CRM), adaptarea la dinamica pieței și măsurarea eficienței sistemului;
 - cerințe cheie: prețuri dinamice și personalizarea termenilor de contractare, gestionarea contractelor încheiate cu partenerii de afaceri, gestionarea conturilor acestora, colaborarea cu partenerii prin logistică și cereri de ofertă, analiza profitabilității afacerii, eficienței economice, eficienței investițiilor, măsurare ROI, etc.

Faza de definire

- Conform metodologiei prezentate **faza de definire** presupune **analiza cerințelor** sistemului de proiectat.
- Fără un set de cerințe (funcționale, nonfuncționale, de implementare, standarde legale, linii directoare, politici și principii de respectat) nu se poate proiecta un sistem care să răspundă necesităților clientilor/beneficiarilor acestuia
- Fără un document de specificații de sistem nu se poate determina dacă soluția corespunde obiectivelor propuse în proiect.

Cerințe funcționale

- Firma are prezență Web și dorește extindere geografică a.î. prin sistemul B2B să dețină o adevărată prezență internațională pe Internet
- Prin implementarea soluției plănuiește o reducere de costuri cu personalul ce asigură service către clienți cât și creșterea gradului de satisfacție al clienților prin asigurarea accesului via Web la produsele firmei, capacitatea de plasare directă a ordinelor, modificare, etc.
- Cerințele funcționale sunt grupate în subsistemele:
 - catalog online,
 - membri,
 - ordine,
 - integrare back-end,
 - service clienți,
 - marketing.

Subsistemul Catalog online

- Caracteristici:

- utilizatorii anonimi pot răsfoi catalogul online dar nu văd prețurile, în scopul de a atrage potențiali clienți, abia după ce se înregistrează pot stabili un contact cu firma;
- posibilitate de căutare a unui anumit produs în catalog fără răsfoire;
- clienții înregistrați și autentificați pot negocia condițiile din contractul încheiat cu firma și pot răsfoi catalogul online în funcție de drepturile acordate,
- permite conectarea cu furnizorii firmei.

Subsistemul Membri

- Caracteristici:
 - noi utilizatori se pot înregistra online (se primește un ID și o parolă după înregistrare),
 - cererile noi de înregistrare sunt mai întâi validate de către administrator clienți (buyer administrator),
 - utilizatorii se pot înregistra o singură dată,
 - informațiile despre clienții sunt introduse într-o bază de date,
 - gestionează profilele clientilor,
 - pentru că se dorește o prezență internațională, site-ul trebuie să suporte mai multe limbi și mai multe unități monetare (informații cerute la crearea profilului clientului).

Subsistemul Membri

- Rolurile client identificate sunt:
 - client anonim (pot doar răsfoi catalogul online și se pot înregistra online),
 - cumpărător (utilizator autentificat ce poate plasa ordine),
 - administrator clienți (validează noile cereri și acordă drepturi de acces),
 - administrator vânzări (validează tranzacțiile clientilor),
 - administrator de sistem.

Subsistemul Ordine

- Caracteristici:
 - permite plasarea ordinelor via Internet,
 - plasarea unui ordin presupune autentificarea anterioară a clientului,
 - se pot plasa orice fel de ordine, în orice stare (activă, inactivă, etc.),
 - se pot completa și ordine pentru care există un formular tip, creat pe baza ordinelor anterior plasate,
 - aprobaarea derulării unui ordin se realizează astfel: aprobaarea inițială de către o persoană autorizată din partea partenerului client (într-o unitate monetară predefinită), această persoană are și dreptul de a configura nivelul de declanșare a derulării ordinului, înregistrarea ordinelor (istoric al introducerii ordinelor) de fiecare dată când este facută, de câte ori este necesar sau când un ordin nu a fost aprobat de persoana autorizată client,
 - clientilor/persoanelor autorizate ce aparțin firmei client li se permite vizualizarea/modificarea stării tuturor ordinelor,
 - datele furnizate către clienți sunt: descrierea produselor din catalog, prețuri, ordinele plasate, starea lor cât și erori de procesare a tranzacțiilor privind ordinele,
 - plata nu se face online.

Subsistemul Integrare back-end

- Caracteristici:
 - actualizarea prețurilor din catalogul online se face prin preluarea datelor din aplicația deja existentă în firmă,
 - actualizarea stocurilor după încheierea unui ordin,
 - descrierea produselor din catalogul online provin din aplicația de gestionare a informațiilor aferente acestora deja existentă,
 - datele privind ordinele plasate online sunt trimise către aplicația de gestionare a ordinelor existentă.

Alte subsisteme

- Subsistemul Service Clienți - Caracteristici:
 - asistență asigurată prin chat,
 - urmărirea interacției client-consultant,
 - pot accesa acest subsistem doar clienții membrii (nu și cei anonimi).
- Subsistemul Marketing - Caracteristici:
 - sunt realizate periodic rapoarte de marketing (activitatea clienților vizavi de produsele firmei, analize de piață, etc.).

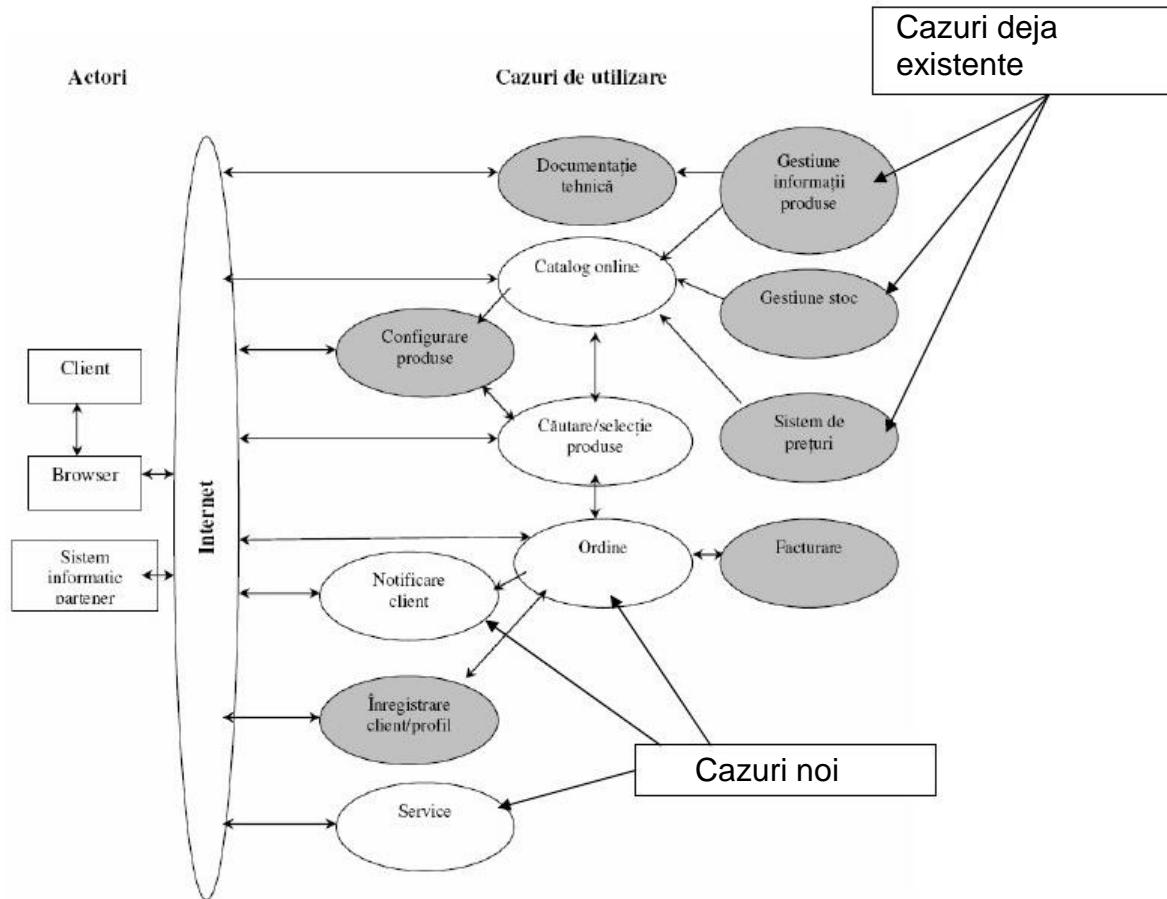
Cerințe non-funcționale

- **Capacitate și performanță:** să accepte maxim 20.000 utilizatori, maxim 30 tranzacții concurente, o medie de 1000 utilizatori, timp de răspuns între 5-10 secunde, să afișeze maxim 2000 de produse, dimensiunea medie a paginii Web dinamice să fie de 50KB și uneia statice de 20KB,
- **Disponibilitate:** 24 de ore pe zi, 7 zile pe săptămână, pe un interval de derulare a afacerilor normal indiferent de fusul orar, disponibilitatea serverelor Web de 90%,
- **Securitate:** autentificare, înregistrare o singură dată a unui utilizator, prelucrarea datelor în timp real sau cel puțin pe loturi de tranzacții, încheierea sesiunii de lucru a unui utilizator după 1h de inactivitate, blocarea contului utilizator după tastarea greșită a ID și parolei de 5 ori, criptarea datelor pentru clienții membrii,
- **Management:** proces de backup săptămânal la sfârșit de săptămână, urmărirea și înregistrarea erorilor de înregistrare/accesare a conturilor client,
- **Scalabilitate:** să permită adăugarea de noi produse în catalog, noi clienți sau zone geografice,
- **Întreținere:** să fie compatibil cu standardele tehnologice IT actuale,
- **Utilizabilitate:** să fie compatibil cu orice tip de browser folosit în prezent,
- **Integrarea datelor și refacerea sistemului în caz de dezastre:** pentru situații normale nu mai mult de 1h, pentru cazuri grave de 12h, pentru serverele Web maxim 150h.

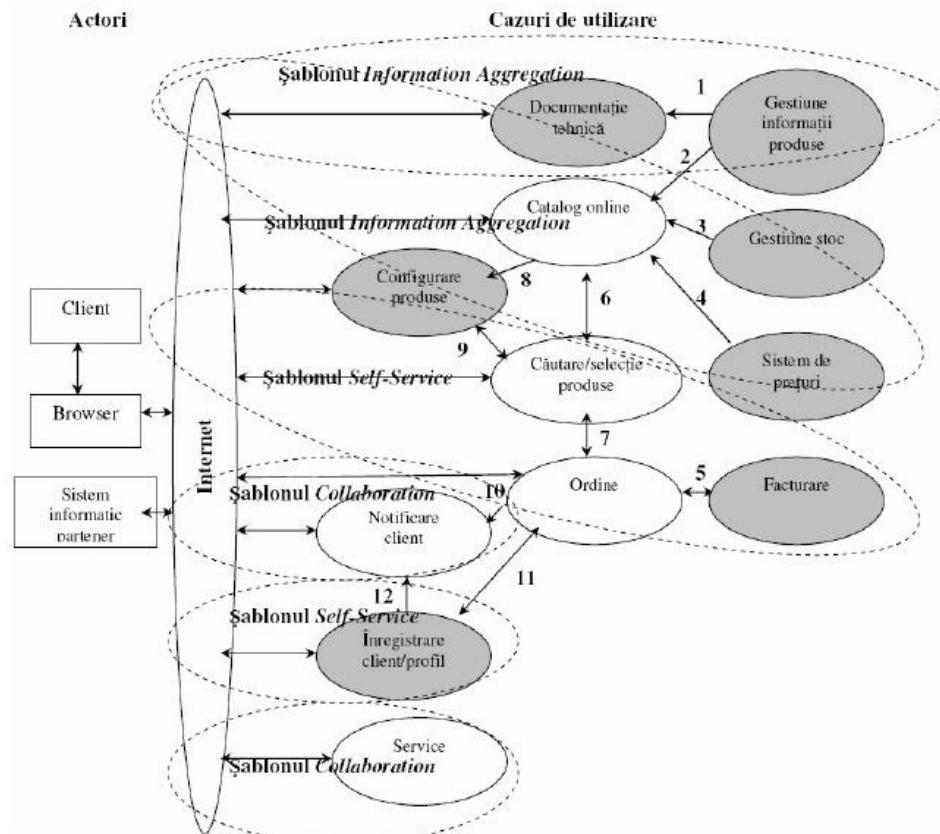
Modelarea proceselor de afaceri

- Soluție de tip highlevel:
 - Clientul utilizează un browser pentru a se conecta la sistem. Dacă nu este înregistrat, se înregistrează, primește un ID și o parolă, prin înregistrare se declanșează și procedura de aprobare/validare din partea persoanei autorizate aparținând firmei client. Abia acum clientul poate să se conecteze la sistemul B2B (login). După conectare un client poate răsfoi catalogul online (use-case) sau documentația tehnică aferentă produselor din catalog (use-case). Catalogul (use-case) furnizează date provenind de la sistemele existente în firma vânzătoare (toate use-case-urile – gestionarea informațiilor despre produse, de stoc și de prețuri) și cazul de utilizare – documentația tehnică a produselor - care preia date din gestiunea informațiilor despre produse. Alte cazuri de utilizare sunt: căutare în catalog, configurare de produse mai complexe (use-case-ul – configurare produs), service (prin chat) interacționând cu personalul consultanță vânzări (actor).
 - După selectarea sau configurarea produsului necesar clientul poate plasa un ordin (use-case) sau poate renunța la acesta (use-case). Sistemul informatic trebuie să valideze ordinele plasate (use-case), după validarea ordinului, actorul – buyer approver – primește notificarea (use-case) prin email.

Actori și cazuri de utilizare pentru sistemul B2B

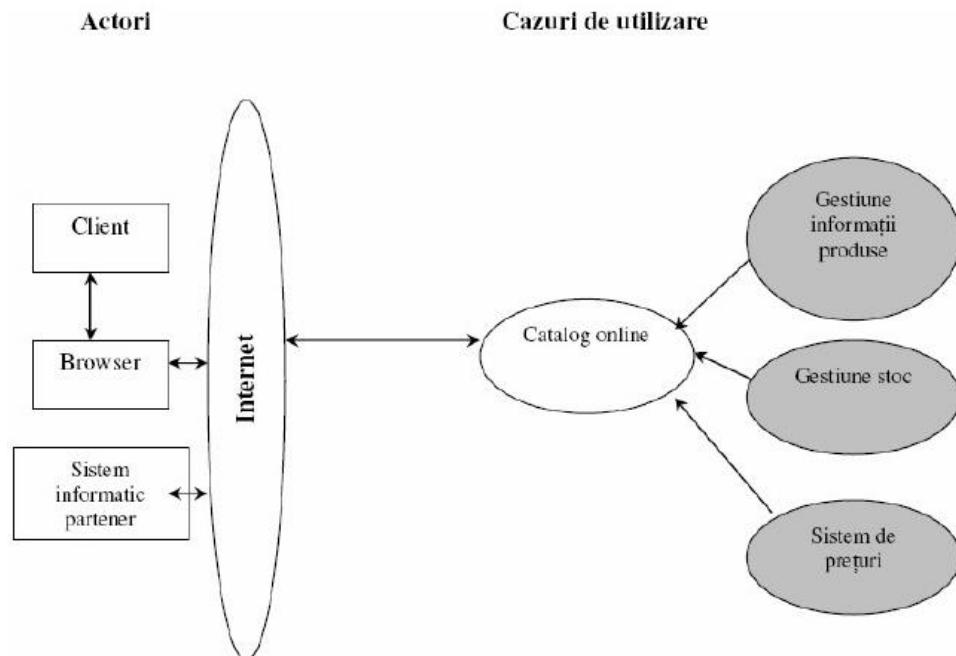


Maparea şabloanelor de analiză pe modelul conceptual al sistemului



Faza de proiectare a aplicației B2B

- Şablonul de integrare a informațiilor provenite din surse diferite



- Un şablon de aplicaţie se obține în urma confruntării cerinţelor sistemului de proiectat cu caracteristicile şablonului de aplicaţie, cât şi cu cerinţele IT de implementare

Tab.7.1.

Obiectivele manageriale	Caracteristicile şablonului de aplicaţie	Population – single step	Population – multi step	Population – crawling and discovery	Population – summarization	Information access – read only
Reducerea costurilor datorită reducerii personalului și fluxurilor de informații necesare gestionării ordinelor de la clienți	Îmbunătățirea eficienței economice	da	da	da	da	da
Scăderii timpului de răspuns end-to-end	Dinamizarea proceselor de afaceri	da	da	da	da	da
Integrarea cu sisteme informatiche existente	Colaborare, transformare și restructurarea datelor provenind din surse diferite		da			

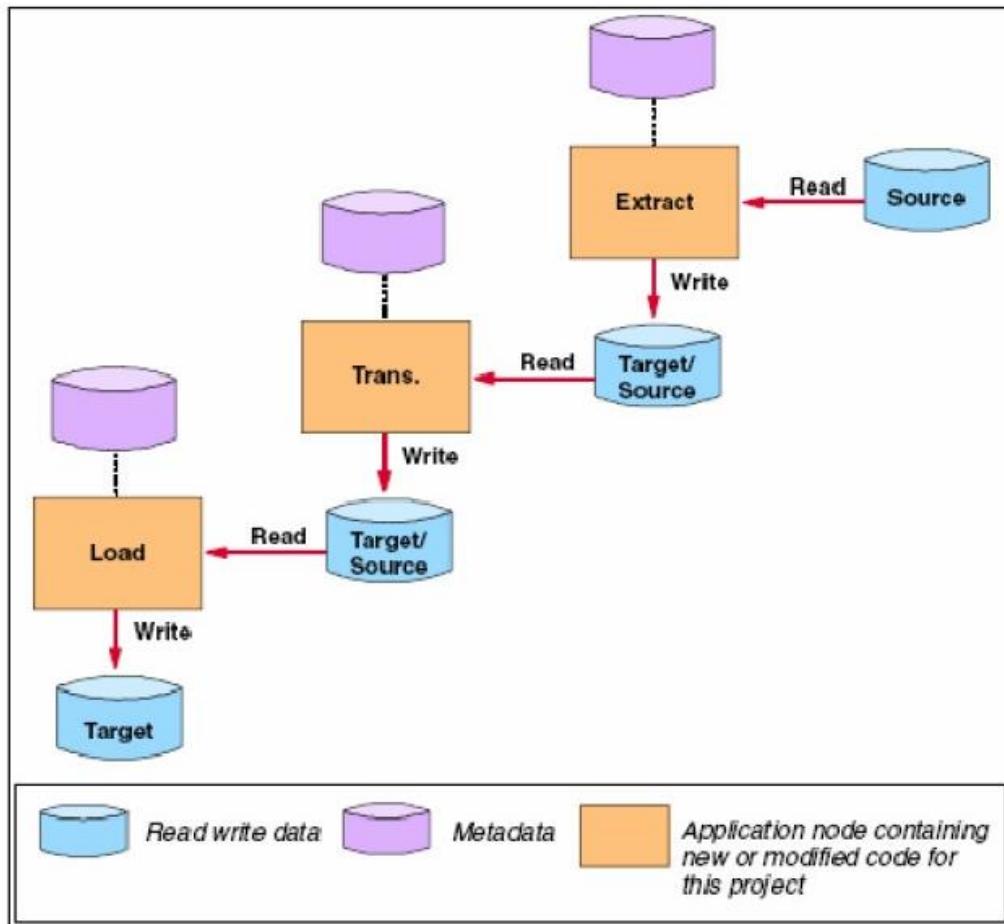
Tab.7.2.

Obiectivele IT	Caracteristicile şablonului de aplicaţie	Population – single step	Population – multi step	Population – crawling and discovery	Population – summarization	Information access – read only
Integrarea cu sisteme informatiche existente	Minimizarea costurilor	da	da	da	da	da
Integrarea cu sisteme informatiche existente	Persistenţa datelor operaţionale	da	da			da
Întreținere facilă	Întreținere	da	da	da	da	da

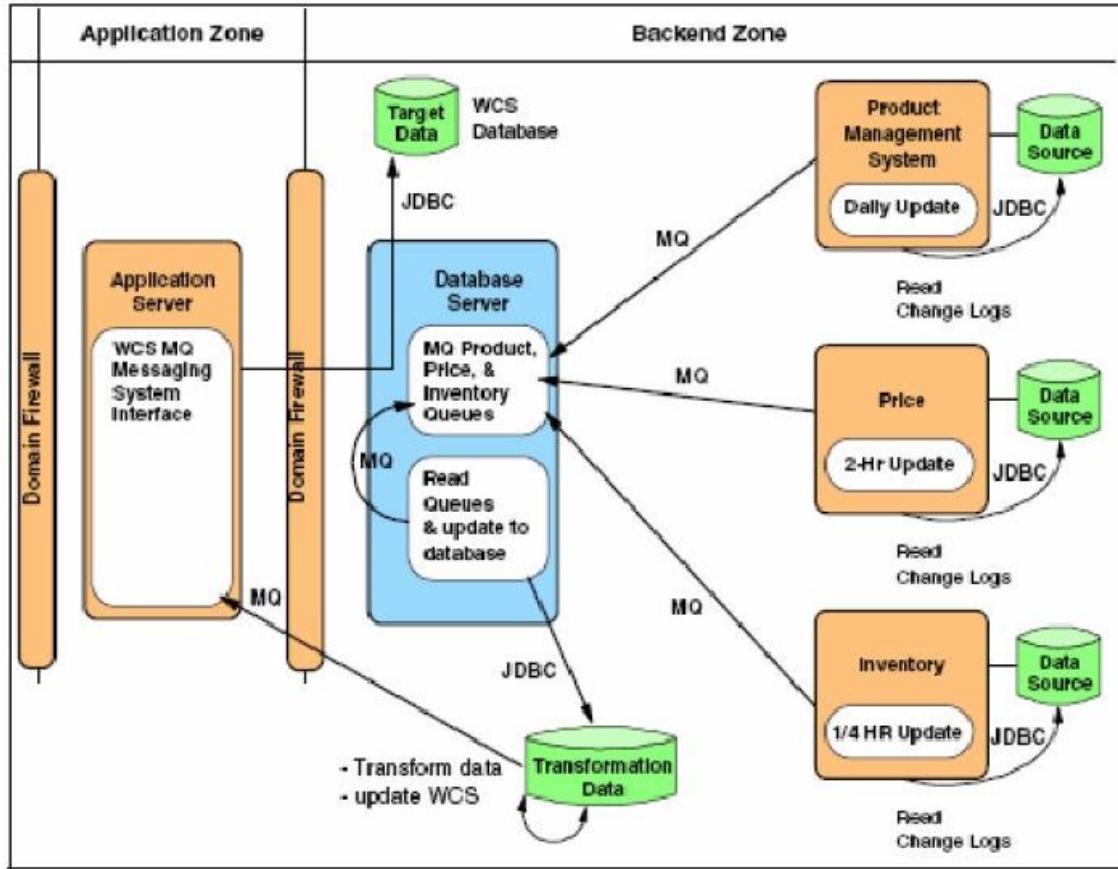
Faza de proiectare a aplicației B2B

- Se impune identificarea regulilor de transformare a datelor ce provin din surse individuale într-o singură direcție – catalogul online.
- Datele furnizate trebuie să fie actualizate (situația stocurilor actualizate în timp real, prețurile, dar și descrierile produselor se pot schimba destul de dinamic) și trebuie transformate înaintea afișării în catalogul online.
- Prin confruntarea obiectivele manageriale și de implementare cu caracteristicile şablonelor de aplicație şablonul ce îndeplinește cel mai bine obiectivele impuse este cel Population – multi step

Şablonul de aplicație Population – Multi Step



Şablonul de execuţie



- Analiză comparativă a procesului de realizare clasic vs. cel bazat pe şabloane - îmbunătățire semnificativă:
se reduce timpul necesar întregului proces de dezvoltare software, crește productivitatea muncii, etc.
- Aplicând metoda top-down și pornind de la modelul componentelor proceselor de afaceri se obține modelul proceselor de afaceri, apoi modelul de domeniu și modelul use-case, iar prin metoda bottom-up se obține modelul sistemelor externe și, în final, se creează modelul serviciilor...

- Modelul componentelor proceselor de afaceri se realizează la nivelul strategic al firmei:

Tab.7.3

	Stoc	Vânzări	Marketing
<i>Procese decizionale</i>	Planificarea aprovisionării	Planificare vânzări	Strategii de preț Strategii de marketing
<i>Procese de control</i>	Urmărire și verificare aprovisionare/livrare	Urmărire și verificare vânzări	Metode/tehnici de marketing
<i>Procese operationale</i>	Livrări Service Reaprovisionare Întreținere	<u>Comenzi online</u>	Prospectarea pieței Promovare Prelucrare date clienți Rapoarte

Bibliografie

1. I.Gh. Rosca, Cristina Bucur, Carmen Timofte, Octavian Paiu, Mirela Visean, Comertul electronic. Concepte, tehnologii și aplicatii, , Ed. Economică, 2004.
2. Janice Reynolds, The Complete E-Commerce Book, Second Edition: Design, Build & Maintain a Successful Web-based Business, 2004
3. Ian Daniel, E-commerce Get It Right!Essential Step-by-Step Guide for Selling & Marketing Products Online. Insider Secrets, Key Strategies & Practical Tips - Simplified for Start-Ups & Small Businesses, NeuroDigital, 2011,
4. Dave Chaffey, E-Business and E-Commerce Management: Strategy, Implementation and Practice (5th Edition), Prentice Hall,2011
5. Thomas M McFayen, eCommerce Best Practices - How to market, sell, and service customers with internet technologies, 2008
6. Larry Ullman, Effortless E-Commerce with PHP and MySQL, New Riders, 2011

http://www.amfiteatrueconomic.ase.ro/arhiva/pdf/no21/articol_fulltext_pag156.pdf

http://www.tutorialspoint.com/e_commerce/e_commerce_quick_guide.htm

http://www.tutorialspoint.com/e_commerce/e_commerce_resources.htm

http://www.tutorialspoint.com/e_commerce/e_commerce_tutorial.pdf