

O abordare duala a realizarii aplicatiilor cu baze de date de tip Internet

Prof. dr. Manole VELICANU

Catedra de Informatica Economica, A.S.E. Bucuresti

Analist prog. Monica Ileana ANGHEL

Directia Informatica si Telecomunicatii, Banca Romaneasca Grup NBG

The actual context of IT development assumes, among other things, use of two technologies, unanimously accepted by experts. These are databases and Java products. Their use for applications development becomes more and more important, simultaneously with the expansion of computer networks, including the Internet. There are many situations which need the integration of both information technologies. Thus, by one side, on developing database applications, the selected solution includes in addition products of Java platform (applet, servlet, JavaBean, BC4J). By the other side, on developing Java applications we need the access of databases (EJB, JDBC, JavaScript, JSP, JDO). In this paper, the illustration of both directions of integration (the duality) refers Oracle system.

Keywords: database, DBMS – database management systems, Java products, Oracle, Internet, network computing.

Adaptarea la noile cerinte ale tehnologiei informatiei si comunicatiilor impune eforturi sustinute din partea proiectantilor si programatorilor de aplicatii, pentru asimilarea noutatilor si integrearea lor în produsele software rezultate. Tehnologia Java este una de mare actualitate în dezvoltarea aplicatiilor cu baze de date de tip Internet. Aceasta, deoarece avantajele majore aduse de stocarea si prelucrarea informatiei din bazele de date sau multiplicat prin facilitatile de comunicare si facilitatile orientate obiect oferite de platforma Java, prin limbajele Java si Javascript, precum si prin alte numeroase interfete dezvoltate in jurul lor.

Aplicatiile cu baze de date de tip Internet își anunta încă din denumire natura dubla, ca urmare a coexistentei celor doua elemente (bazele de date si comunicatiile). Abordarea duala mentionata in titlu, vizeaza o prezentare sintetice a celor doua modalitati de dezvoltare software: aplicatii cu baze de date ce includ tehnologii Java si acces din platforma Java la bazele de date.

Utilizarea tehnologiilor Java la realizarea aplicatiilor cu baze de date

Larga utilizare a bazelor de date a necesitat implementarea unor facilitati pentru lucrul în rețeaua de calculatoare globala. De aceea,

sistemele de gestiune a bazelor de date, în versiunile lor din ultimii ani, permit utilizarea peste Internet a aplicatiilor realizate cu ajutorul lor. Acest fapt este posibil si datorita încorporarii unor elemente din platforma Java în mediul de lucru specific bazelor de date.

Produsele Oracle, multa vreme concentrate asupra dezvoltarii facilitatilor pentru bazele de date relationale, în ultimii ani s-au îmbogățit cu multe componente necesare construirii relativ usoare a unor aplicatii robuste în mediu distribuit, ca si pentru comerțul electronic. Vom enumera în continuare câteva produse ale platformei Java implementate în mediul Oracle, pentru care se va regasi si o sintetica prezentare. Astfel, cea mai folosita cale de încarcare a bazelor de date, lucrul cu formulare (forme), de sub Oracle, respectiv Oracle Forms Developer, permite desfasurarea pe Web, prin descarcarea fisierelor de clase Java pe browser-ul clientilor. Aceste fisiere de clase constituie interfata cu utilizatorul pentru modulele aplicatiilor Forms Server si întretin comunicatia între server si browser-ul client.

Applet-urile sunt componente scrise în cod Java, inclus într-un fisier HTML. Ele sunt interpretate la PC-ul gazda de catre browser, gratie faptului ca acesta colaboreaza cu masina virtuala Java. Acestea sunt responsabile

cu descărcarea și executia fișierelor de clase Java care creează interfața cu utilizatorul pentru aplicațiile Oracle Forms Developer. La momentul executiei, fișierele de clase Java sunt prelucrate fie de mașina virtuală Java (JVM), inclusă în browser-ul Web, fie într-un modul Applet Viewer. Java Runtime Environment (JRE) este o combinație între JVM și fișierele suport. Oracle JInitiator oferă o alternativă la browser-ul standard JRE, prin îmbunătățirea desfășurării pe Web a aplicațiilor Oracle Forms. El este oferit sub forma unei adăugiri (*plug-in*) la Netscape, sau ca o componentă ActiveX la Microsoft Internet Explorer.

Servlet-urile denumesc clase (module script) Java, necesare construirii unor pagini Web dinamice, încărcate și menținute rezidente pe server. Avantajele aduse de această tehnologie constau în faptul că serverele Web pot să se conecteze și să prelucreze informația în mod dinamic, să ruleze aplicații Web și nu doar să transfere documente statice. Soluția menține executabilul persistent pe server, între cererile clienților, eliminând dezavantajele programării Common Gateway Interface (CGI), unde fiecare cerere a clientului lansează un nou proces pe server. Se recomandă utilizarea acestei tehnologii pentru aplicațiile care accesează baze de date (serverul realizează accesul la date, clienții doar formulează cererile de regăsire).

Java Bean-urile sunt programe Java independente de platformă, portabile și reutilizabile. Ele ajută la modificarea apariției și comportamentului interfeței cu utilizatorul, prin includerea în aplicațiile create cu Oracle Form Builder. Pentru a include asemenea elemente într-un formular este necesar ca în cadrul sau să fie definite obiecte de tip Bean Area, care permit gazduirea de componente Java Bean în aplicațiile desfășurate pe Web. În paleta de proprietăți a obiectelor Bean Area se găsește una (Bean Area Implementation Class) a cărei introducere este obligatorie și care va conține numele fișierului cu clasa Java importantă pentru economia aplicației. Întrucât Java este un limbaj *case-sensitive*, numele fișierului de clase va fi specificat exact, inclusiv calea completă. După specifi-

carea proprietății Implementation Class, modulul Java Bean-ul va fi vizibil în Bean Area. Acesta inițiază comunicatia cu formularul prin declansarea unui eveniment client (Custom Event), care trimite un nume de eveniment și numele unei liste de parametri conținând datele asociate la modulul formular. Această informație se obține cu ajutorul a două variabile de sistem: una definește numele evenimentului tratat în Java Bean, cealaltă numele listei de parametri care conține datele evenimentului asociat. Ori de câte ori Java Bean comunică cu formularul, se lansează un declansator (When-Custom-Item-Event) care trebuie atașat la Bean Area. Prin examinarea variabilelor de sistem în cadrul trigger-ului se poate identifica ce eveniment s-a produs și se poate aplica acțiunea corespunzătoare.

De la versiunea Oracle 8i de bază de date, această conține clase și proceduri stocate Java alături de cele PL/SQL. De asemenea, permite și dezvoltarea de Enterprise Java Bean (EJB) care, spre deosebire de Java Bean-uri, nu rezolvă funcții de interfață cu utilizatorul, ci încapsulează și distribuie logica Java între mai multe servere, în scopul de a gestiona datele, securitatea tranzacțiilor și resurselor.

BC4J (Business Component 4 Java) este un cadru general al aplicațiilor, pentru generarea de logică a afacerilor reutilizabilă. Pot fi încapsulate reguli de afaceri în biblioteci de componente Java care se pot reutiliza prin generarea de viziuni bazate pe SQL. De exemplu, cu obiectele Oracle 8i XML se pot construi aplicații de comerț electronic bazate pe mesaje, care implementează reguli de afaceri specifice.

Dezvoltarea de proceduri stocate Java, EJB și BC4J poate fi realizată comod cu Oracle JDeveloper, dar sunt necesare cunoștințe de programare în Java. Odată cu creșterea utilizării Java în industria IT a crescut și numărul de puncte de integrare cu Java pentru tot mai multe aplicații. Astfel și numărul de biblioteci standard pentru aplicațiile de întreprindere este în dezvoltare.

De la versiunea 2 a lui Oracle Forms 6i a apărut o nouă caracteristică, Java Importer,

care face mai usoara invocarea logicii afacerii, scrisa in Java, din aplicatia Forms. Prin aceasta componenta pot fi generate automat pachete si proceduri PL/SQL care permit crearea si utilizarea obiectelor Java direct în aplicatia Forms, prin intermediul celor trei elemente ale sale: *generatorul* (creeaza pachetele PL/SQL pentru accesul la clasa specificata), *pachetul ORA_JAVA* (furnizeaza un set de functii pentru asistarea lucrului cu pachetele selectate) si *puntea JavaNative Interface*, care gestioneaza interactiunea cu clasele Java la momentul executiei.

Accesul din platforma Java la bazele de date

Daca în cele de mai sus am prezentat modalitati de desfasurare a aplicatiei cu baze de date pe Internet, sa ne oprim putin si asupra accesului de sub platforma Java la bazele de date.

● Enterprise Java Bean (EJB) sunt componente de nivel mediu care încapsuleaza procesele de afaceri si ofera servicii la distanta clientilor. Modulele EJB pot fi de tip:

- *session*, orientate pe sarcini (în varianta *stateless* proceseaza o singura cerere, iar în varianta *stateful* mentine stările);

- *message driven*, mapeaza mesaje, fara sa existe stare;

- *entity*, module mapate pe tabelele bazelor de date, care reprezinta informatii din orice tip de sursa de date si asigura persistenta datelor cu mecanismele Customer Managed Persistence (CMP) si Bean Managed Persistence (BMP).

Dintre avantajele aduse de EJB, promotorii IDE JDeveloper amintesc: arhitectura de desfasurare standard, persistenta, gestionarea tranzactiilor (facilitati de Commit/Rollback) si administrarea mai buna a conexiunilor cu un mecanism Connection Pooling, prin care conexiunea este dezactivata la încheierea *task*-ului pentru care a fost definita.

● Java Database Connectivity (JDBC) este un produs tip API Java pentru accesul la orice tip de tabela, asigurând patrunderea aplicatiei Java la orice sursa de date, independent de baza de date, cu conditia sa fie disponibile drivere JDBC corespunzatoare, care sa îndeplineasca urmatoarele functii:

- stabilirea conexiunii la baza de date;
- transmiterea interogarilor SQL si a instructiunilor de actualizare la sursa de date;
- posibilitatea procesarii rezultatelor.

Tehnologia JDBC este realizata prin utilizarea unei biblioteci de clase (*java.sql*) care permite executarea unei cereri de tip SQL pentru baze de date relationale, din aplicatii scrise în Java. Avantajul utilizarii acestui pachet este ca aplicatiile Java sunt aceleasi, indiferent de baza de date. În acest fel *applet*-urile pot sa interogheze bazele de date indiferent de tipul lor. Cele mai importante clase din biblioteca *java.sql* sunt:

- *DriverManager*, folosita pentru a facilita utilizarea mai multor *drivere* de baze de date într-o singura aplicatie, fiecare dintre *drivere* putând fi utilizat pentru conectarea la o sursa diferita de date;

- *Connection*, obiect care specifica cu ce baza de date se face comunicatia;

- *Statement*, o instructiune SQL aplicata bazei de date.

● *JavaScript* este o componenta din setul de instrumente al programatorilor Web. Fiind un limbaj, depinde pentru executie de *browser*. JavaScript ofera capacitatea de a face ca paginile statice sa devina interactive si mai prompte fata de actiunile utilizatorilor si de intrarile acestora, prin înglobarea unui *script* în structura HTML existenta. Tehnologia cuprinde obiecte specifice bazelor de date, cum ar fi:

- *Database*, încapsuleaza toata functionalitatea asociata interactiunii cu o baza de date relationala. Câteva dintre cele mai importante metode care caracterizeaza acest obiect sunt: *beginTransaction()*, *endTransaction()*, *commitTransaction()*, *connect()*, *cursor()*, *disconnect()*, *rollbackTransaction()*;

- *DbPool*, reprezinta conexiuni la baza de date;

- *Connection*, refera o conexiune data, atunci când este extrasa dintr-un ansamblu de conexiuni;

- *ResultSet*, stocheaza rezultatul rularii unei proceduri stocate;

- *Cursor*, creat prin apelarea metodelor *connection.cursor* sau *database.cursor*;

- BLOB, asigura suport pentru lucrul cu obiecte binare mari.

• Java Server Pages (JSP) este o interfata pentru construirea aspectelor estetice ale paginilor Web (amplasare, aspect), mai putin utila ca sursa si manipulare a datelor dinamice. Dintre caracteristicile sale mentionam:

- poate functiona pe o multitudine de servere;
- separa logica aplicatiei (în ODBC, Java Bean s.a) de prezentarea acesteia (în JSP);
- permite reutilizarea unor componente generate cu alte tehnologii (JavaBean) atât independent, cât si în cadrul unor interfete de dezvoltare a paginilor Web;
- este independenta de platforma pe care ruleaza (script-urile fiind bazate pe Java);
- se poate utiliza în aplicatii multinivel;
- executia unei pagini JSP este realizata de motorul JSP instalat pe un server de Web, care proceseaza pagina JSP (o pagina HTML cu elemente aditionale) si returneaza o pagina HTML/XML

• Java Data Objects (JDO) este un produs API pentru accesul transparent la bazele de date. Programatorul poate scrie cod Java care acceseaza transparent datele, fara sa scrie cod specific SGBD-urilor. Aceasta este o tehnologie complementara celei JDBC. Punctul sau forte este ca standardizeaza obiectele bazei de date si maparile obiectual-relationale pentru limbajul de programare Java, permitând programatorului sa foloseasca clase din limbajul Java în loc sa copieze date aparținând diferitor modele de date. JDO si JDBC sunt abordari complementare. Ambele tehnologii au propriile puncte forte si pot fi utilizate de programatori cu diferite aptitudini. JDO îi permite programatorului sa acceseze baze de date fara a avea cunostinte de SQL.

Bibliografie

- [VELU03] – Manole Velicanu, Ion Lungu s.a - Sisteme de baze de date – teorie si practica, Ed. Petron, Bucuresti, 2003
- [SOCA03] – Dan Somnea, Mihai Calciu - Ghidul managerului pentru noile tehnologii informatice si de comunicatie, Editura ASE, 2003
- [VELU02] – M. Velicanu, I. Lungu, M. Muntean, M. Iorga, S. Ionescu - ORACLE – Platforma pentru baze de date, Ed.Petron, Bucuresti 2002
- [ROSC02] – Rosca Ioan - Intranet, Editura ASE, 2002
- [WAWY01] – Richard Wgner, R.Allen Wyke - JavaScript, Editura Teora, 2001
- [ORAC01] – Oracle iDS Forms: Build Internet Applications 400346811, Production 1.1, June 2001, Oracle
- [FOSM01] – Ben Forta, Edwin Smith, S.Stirling, L. Kim, R.Kerr, D.Aden, A.Lei - Dezvoltarea aplicatiilor Java Server Pages, Ed. Teora, 2001
- [NOST97] – Peter Norton, William Stanek - Ghid de programare in Java, Editura Teora, 1997
- www.otn.oracle.com situl Oracle Tutorial Network
- www.sun.com