

Komercyjne bazy danych 2

Oracle a technologie internetowe

doc. dr inż. Tomasz Traczyk

Instytut Automatyki i Informatyki Stosowanej
Politechnika Warszawska

Styczeń 2010

Oracle Fusion Middleware

- ▶ Zbiór produktów służących do budowy aplikacji w architekturach wielowarstwowych i/lub rozproszonych (SOA)
- ▶ Zawiera m. in.
 - ▶ Serwery aplikacyjne z kontenerami JEE (*Java Enterprise Edition*)
 - ▶ Narzędzia do budowy dynamicznych stron internetowych
 - ▶ Narzędzia do integracji systemów, w tym dla SOA (*Service Oriented Architecture*)
 - ▶ Narzędzia CMS (*Content Management Systems*)
 - ▶ Narzędzia BI (*Business Intelligence*) i raportowania (np. *BI Publisher*)
 - ▶ Systemy do budowy portali
 - ▶ Narzędzia do uwierzytelniania/autoryzacji i zarządzania tożsamością (w tym LDAP)
 - ▶ Narzędzia do zarządzania systemami i optymalizacji wydajności
- ▶ Możliwości budowania architektur klastrowych i gridowych
- ▶ Na skutek mało przejrzystej polityki oraz licznych zakupów/przejęć zbiór narzędzi jest mocno redundantny

Oracle WebLogic Server

- ▶ Rozbudowany serwer aplikacyjny
- ▶ Zakupiony przez Oracle wraz z firmą BEA
- ▶ Zastąpił produkty
 - ▶ Oracle Application Server – spreparowany Apache z dodatkami
 - ▶ Oracle Components for Java (OC4J) – kontener Java EE
- ▶ Przeznaczony dla klientów korporacyjnych

„Wagi” Oracle WebLogic Server

- ▶ WLS Standard Edition – wersja podstawowa
 - ▶ kontener technologii JEE
 - ▶ monitor transakcyjny Tuxedo
- ▶ WLS Enterprise Edition – wersja do budowy dużych instalacji: rozwiązania klastrowe (*failover*, migracja usług itp.), rozwiązanie dla dużej wydajności (*caching*), zarządzania i diagnostyczne
- ▶ WebLogic Suite – pełna wersja: dodatkowo serwery Forms i Reports, Portal, Discoverer, maszyna wirtualna JRockit JVM czasu rzeczywistego, rozwiązania dla SOA, rozwiązania gridowe

Narzędzia deweloperskie wspierające WLS

- ▶ Oracle JDeveloper
- ▶ Oracle Enterprise Pack for Eclipse – wtyczka do Eclipse

Technologie Java w Oracle WLS (wybór)

- ▶ WebLogic – renomowany kontener JEE (zakupiony od firmy BEA)
- ▶ Własny ORM (*Object-Relational Mapper*)
 - ▶ *Oracle Top Link* – z pełnym wsparciem dla możliwości i typów danych bazy Oracle, także dla XML
 - ▶ *Oracle Top Link Essentials* – ograniczone wsparcie dla zaawansowanych cech bazy, ale pełna zgodność z JPA (*Java Persistence API*)

WebLogic

Możliwości (wybór)

- ▶ Java EE we w miarę nowym wydaniu
- ▶ Kontener serwletów, JSP i JSF
- ▶ *Enterprise Java Beans 3.0*
- ▶ *Web Services*
- ▶ wsparcie dla technologii .Net
- ▶ wsparcie dla technologii Struts i Spring
- ▶ itd.

Bramki PL/SQL

- ▶ Pozwalają wywoływać procedury PL/SQL przez HTTP i adres URL
- ▶ Parametry do procedur PL/SQL są przekazywane metodą GET (w URL) lub POST
- ▶ Biblioteki PL/SQL Web Toolkit pozwalają
 - ▶ wysyłać wyniki działania procedur jako *response* przez HTTP
 - ▶ łatwo formatować odpowiedź w HTML
- ▶ Udostępniają LOB-s z bazy danych tak, jak gdyby były one plikami na serwerze (np. do użycia w ``)
- ▶ Postać URL¹:

`http://host[:port]/virthpath/[!][schema.][package.]procname[?params]`

¹Znak ! umożliwia przekazywanie zmiennej liczby parametrów przez tabele nazw i wartości

mod_plsql

- ▶ Bramka zrealizowana jako moduł serwera Apache
- ▶ Dostarczana wraz z:
 - ▶ serwerem Oracle HTTP Server – dostarczany jako część składowa instalacji bazy danych Oracle do wersji 10g
 - ▶ serwerem Oracle Application Server – poprzednikiem WLS
- ▶ Konfigurowana przez odpowiednie pliki konfiguracyjne `.conf` (Apache-podobne) i/lub przez interfejs WWW
- ▶ Dostęp do bazy danych definiują tzw. DAD-s (*Database Access Descriptors*)
- ▶ Istnieją odpowiedniki *open-source*, możliwe do zainstalowania np. w serwerze Apache

Database Access Descriptor (DAD)

- ▶ Definiuje wirtualny adres wywołania procedury
- ▶ Określa m. in.
 - ▶ sposób połączenia z bazą: instancję, użytkownika i ew. hasło
 - ▶ procedury do uploadu i downloadu plików
 - ▶ procedurę dodatkowo zabezpieczającą dostęp
 - ▶ stronę kodową odpowiedzi HTTP

Embedded PL/SQL Gateway

- ▶ Bramka HTTP będąca częścią XML DB
- ▶ Może służyć także jako bramka PL/SQL
- ▶ Korzysta z listenera Oracle Net, ale nasłuch nie jest konfigurowany w pliku `listener.ora`, tylko wewnątrz bazy danych
- ▶ Konfigurowana wewnątrz bazy danych za pomocą pakietu `DBMS_EPG`
 - ▶ konfigurację można wyświetlić za pomocą skryptu `pepgstat.sql`
- ▶ Wada: działa na komputerze, na którym pracuje DBMS, więc profesjonalne instalacje wymagają użycia proxy

ApEx Listener

- ▶ **Nowość!**
- ▶ Napisana w języku Java darmowa (!) bramka PL/SQL specjalizowana do wykorzystania z Oracle Application Express
- ▶ Daje się instalować na serwerach Oracle WLS, Oracle AS+OC4J, Tomcat
- ▶ Możliwości podobne do `mod_plsql`, ale jedna instancja obsługuje tylko jedno połączenie (odpowiednik DAD)
- ▶ Wbudowane buforowanie plików (*cache*)
- ▶ Konfiguracja za pomocą pliku konfiguracyjnego w XML i/lub interfejsu WWW
- ▶ Zawiera dodatkowo program Apache FOP, skonfigurowany do przekształcania raportów ApEx 4.0 na PDF

PL/SQL Web Toolkit

- ▶ Biblioteka PL/SQL umożliwiająca dynamiczne budowanie stron w (X)HTML lub XML i wysłanie ich jako *response* przez HTTP

Pakiety (wybór)

- ▶ `http` – formuje strony HTML oraz umożliwia wysłanie odpowiedzi przez HTTP
- ▶ `htf` – odpowiednik `http`, ale wyników nie wysyła przez HTTP, tylko zwraca w postaci funkcji, służy do zagnieżdżenia w wywołaniach `http`
- ▶ `owa_cache` – steruje buforowaniem wyników
- ▶ `owa_cookie` – wspomaga manipulacje ciasteczkami
- ▶ `owa_image` – wspomaga budowanie aktywnych map
- ▶ `owa_util` – różne podprogramy pomocnicze
- ▶ `wpg_docload` – procedury umożliwiające „plikowy” *download* danych z kolumn typu LOB

PL/SQL Web Toolkit, cd.

Budowanie stron WWW za pomocą `http/htf`

- ▶ Procedura `http.p` wysyła tekst jako fragment odpowiedzi HTTP
- ▶ Procedura `http.print` wysyła tekst jako linię odpowiedzi HTTP
- ▶ Specjalizowane procedury pakietu `http` służą do budowy poszczególnych elementów stron (X)HTML, np.
 - ▶ `http.htmlopen/http.htmlclose`, `http.bodyopen/http.bodyclose` itp.
 - ▶ `http.title`, `http.meta`, `http.style` itp.
 - ▶ `http.header`, `http.para`, `http.div` itp.
 - ▶ `http.olistopen`, `http.listitem`, `http.olistclose` itp.
 - ▶ `http.anchor`, `http.br` itp.
 - ▶ `http.tableopen`, `http.tablerowopen`, `http.tabledata` itd.
 - ▶ `http.formopen`, `http.formtext`, `http.formsubmit` itd.
 - ▶ `http.bold`, `http.em`, `http.italic`, `http.small` itp.
- ▶ Funkcje `htf` służą do zagnieżdżenia w wywołaniach `http`, np.


```
http.tablerowopen;
http.p(htf.format_cell(liczba, '999G990D00'));
...
```

PL/SQL Web Toolkit – przykład

```
create or replace procedure uzytkownicy
(wzorzec in varchar2 default NULL) as
    dummy boolean; clause varchar2(1000);
begin
    http.htmlopen; http.headopen;
    http.title('Użytkownicy bazy danych');
    http.headclose;
    http.bodyopen; http.header(1, 'Użytkownicy bazy danych');
    clause := 'where username like upper(''%' || wzorzec
              || '%') order by username';
    dummy := owa_util.tableprint(ctable=>'all_users', cclauses=>clause);
    http.bodyclose; http.htmlclose;
end uzytkownicy;
```

procedura z parametrem ograniczającym wynik

Dodatkowa klauzula ustala selekcję i sortowanie, funkcji `upper` użyto, gdyż nazwy w słowniku danych są zapisywane dużymi literami

Procedura `tableprint` wyprowadza zawartość tabeli/perspektywy w postaci tabeli HTML

Wywołanie np.: `http://host/pls/dad/uzytkownicy?wzorzec=kowal`

PL/SQL Server Pages (PSP)

- ▶ Szablony dynamicznych stron WWW zbudowane podobnie do JSP
- ▶ Ładowane do bazy danych programem `loadpsp` – automatycznie zamieniane na czysty kod PL/SQL, np.


```
set NLS_LANG=AMERICAN_AMERICA.EE8MSWIN1250
loadpsp -replace -user user/password@connect.string plik.psp
```
- ▶ Wywoływane przez bramkę PL/SQL
- ▶ Lepsze od pisania stron wprost w PL/SQL o tyle, że pozwala częściowo oddzielić logikę od prezentacji

Znaczniki PSP

<%@ ... %>	Dyrektywy
<%! ... %>	Deklaracje
<% ... %>	Bloki kodu
<%= ... %>	Wyrażenia
<%-- ... --%>	Komentarze

PL/SQL Web Pages – przykład

```

<%@ page language="PL/SQL" %> <%@ page contentType="text/html" %>
<%@ plsql procedure="uzytkownicy_psp" %>
<%@ plsql parameter="wzorzec" default="null" %>
<%! CURSOR c IS SELECT username FROM all_users
WHERE username like '%'|| upper(wzorzec) ||'%' ORDER BY username; %>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html">
  <title>Lista użytkowników</title>
</head>
<body>
  <h1>Lista użytkowników</h1>
  <table width="40%" border="1">
    <tr><th>Użytkownik</th></tr>
    <% FOR i IN c LOOP %>                                pętla
    <tr><td>
      <%= i.username %>                                  użycie zmiennej
    </td></tr>
    <% END LOOP; %>
  </table>
</body>
</html>

```

Systemy do budowy portali

- ▶ Służą do budowania portali „z klocków” zwanych portletami
- ▶ Zawierają CMS z odpowiednim repozytorium
- ▶ Przeznaczone raczej do budowy dużych systemów o intensywnym wykorzystaniu, głównie klasy EIP (*Enterprise Information Portals*) – portali wewnątrzorganizacyjnych (do użycia w intra/ekstranecie)
- ▶ Oferta Oracle
 - ▶ Oracle Portal
 - ▶ Oracle WebLogic Portal (wcześniej BEA WebLogic Portal)
 - ▶ WebCenter Suite

Oracle Portal

- ▶ Narzędzie do budowy portali
- ▶ System CMS (*Content Management System*)
- ▶ Z możliwością budowania stron dynamicznych z danymi z bazy oraz niewielkich aplikacji do pracy z danymi
- ▶ Zintegrowany z Oracle Text – duże możliwości wyszukiwania pełnotekstowego

Składniki portalu

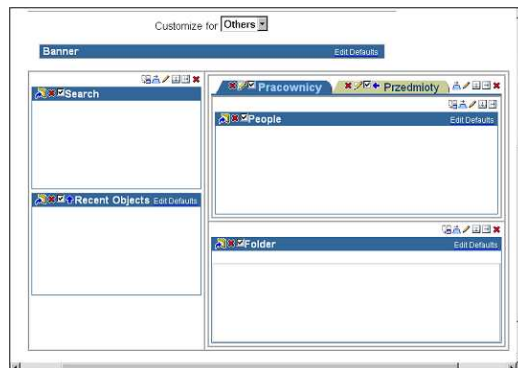
- ▶ Strony
- ▶ Portlety
- ▶ Obszary treści
- ▶ Aplikacje

Strony (*pages*)

- ▶ Są interfejsami do portalu
- ▶ Stanowią „centra informacyjne”
- ▶ Składają się z obszarów, zawierających portlety
- ▶ Każdy użytkownik ma przypisaną stronę startową (*home page*)
- ▶ Cechy wizualne
 - ▶ styl strony (*style*): kolory, czcionki, tło
 - ▶ układ strony (*layout*)
- ▶ Strony tytułowe (*top-level*) – wyróżnione

Układ strony (*layout*)

- ▶ Nagłówek
- ▶ Obszary (*regions*) – elastyczny podział strony
- ▶ Zakładki – w obszarach, można zagnieździć



Portlety (*portlets*)

- ▶ Przygotowane elementy mające postać stron WWW
 - ▶ dokumenty (np. HTML)
 - ▶ aplikacje
- ▶ Mogą być umieszczone w prostokątnym obszarze innej strony
- ▶ Są „cegiełkami” z których składane są strony
 - ▶ każda strona składa się z portletów
 - ▶ portlety są przeznaczone do użycia na wielu stronach
- ▶ Strony mogą pełnić rolę portletów dla innych stron
- ▶ Portlety mogą być dostarczane spoza systemu Oracle Portal
 - ▶ przez zarejestrowanych dostawców (*portlet providers*)
 - ▶ z zarejestrowanych węzłów

Portlety użytkownika

- ▶ Perspektywy, kategorie itp.
- ▶ Strony użytkownika
- ▶ Aplikacje
- ▶ Portlety i aplikacje zewnętrzne

Portlety predefiniowane

- ▶ Ogólne: login, spis użytkowników
- ▶ Wyszukiwanie: w obszarach treści, w systemie Portal, zapamiętane wyszukiwanie
- ▶ Do zarządzania treścią: ulubione, ostatnie obiekty, foldery użytkownika, łączniki do aplikacji zewnętrznych
- ▶ Do budowania stron (tworzenie i edycja): strony, aplikacje Oracle, HTML
- ▶ Administrowanie systemem Portal: użytkownicy, grupy, węzły, dostawy, usługi zewnętrzne
- ▶ Administrowanie bazą danych
- ▶ Monitorowanie: systemu Portal, bazy danych

Obszary treści (*content areas*)

- ▶ Służą do przechowywania i udostępniania dokumentów
- ▶ Uprawnieni użytkownicy mogą samodzielnie umieszczać w nich dokumenty
- ▶ Są zorganizowane za pomocą folderów
- ▶ Kategoryzacja w ramach obszaru: kategorie, perspektywy
- ▶ Cechy obszarów: prawa dostępu, typ wersjowania
- ▶ Użycie obszarów treści
 - ▶ „Szkielet” obszaru tworzy projektant portalu
 - ▶ Uprawnieni użytkownicy umieszczają elementy i foldery

Type	Name	Actions	Owner	Modified
Folders	Folders	Create	SCOTT	21-Nov-2001
Categories	Categories	Create	SCOTT	21-Nov-2001
Navigation Bars	Navigation Bars	Create	SCOTT	21-Nov-2001
Perspectives	Perspectives	Create	SCOTT	21-Nov-2001
Styles	Styles	Create	SCOTT	15-Nov-2001
Custom Types	Custom Types		SCOTT	21-Nov-2001

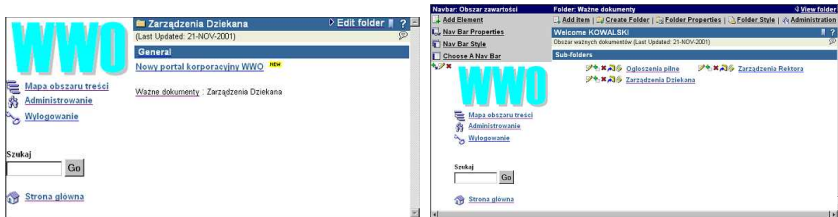
Obszary treści: foldery

- ▶ Tworzą zagnieżdżane struktury
- ▶ Rodzaje folderów: kontener, URL, wynik wyszukiwania (*pre-defined search*), PL/SQL – strony dynamiczne Web PL/SQL, typu użytkownika
- ▶ Mogą mieć: zróżnicowane prawa dostępu, zróżnicowane wersjowanie, styl wyglądu



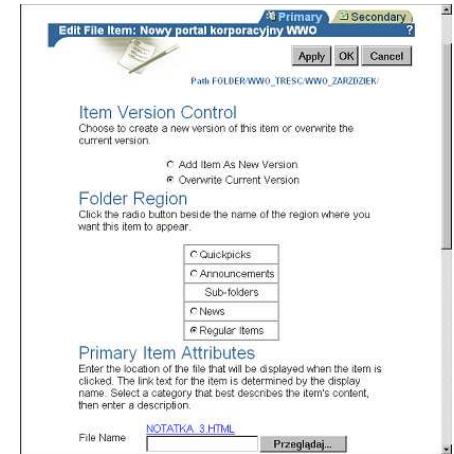
Styl folderów

- ▶ Nagłówki (*banners*, kilka poziomów)
- ▶ Kolory, czcionki
- ▶ Układ obszarów: *quickpicks*, *announcements*, *sub-folders*, *news*, *regular items*
- ▶ Obszar nawigacji (*navigation bar*)



Obszary treści: elementy

- ▶ Przechowują treść
- ▶ Typy elementów: plik, tekst, URL, plik ZIP, łącznik do folderu, typ użytkownika
- ▶ Standardowe atrybuty elementów: data, autor, słowa kluczowe, okres ważności, sposób wyświetlania



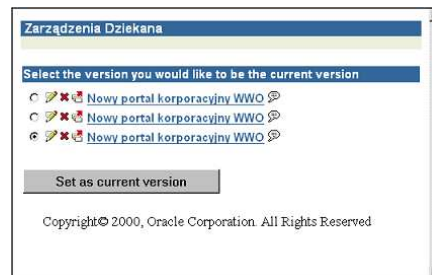
Obszary treści: atrybuty i wersje

Atrybuty i nowe typy

- ▶ Można zdefiniować dodatkowe atrybuty
- ▶ Atrybuty te można przypisać nowym typom folderu lub elementu (z wartością domyślną)

Wersjowanie

- ▶ Rodzaje
 - ▶ *none* – bez wersjowania
 - ▶ *simple* – użytkownik decyduje czy zachować starą wersję
 - ▶ *audit* – „przymusowe” zachowanie starej wersji
- ▶ *Check-out / check-in* – zabezpieczenie przed równoległą modyfikacją



Kategoryzacja

Kategorie

- ▶ Przeznaczenie
 - ▶ sztywny podział tematyczny treści
 - ▶ folderów
 - ▶ elementów
 - ▶ podział pokazany na stronach
- ▶ Cechy
 - ▶ każdy obiekt musi należeć do kategorii
 - ▶ każdy obiekt należy do jednej kategorii
 - ▶ kategorie są określone w ramach obszaru treści
- ▶ Wykorzystanie
 - ▶ do wyszukiwania
 - ▶ do użycia jako portlet

Kategoryzacja

Perspektywy

- ▶ Przeznaczenie
 - ▶ elastyczny podział treści na obszary zainteresowań
 - ▶ zwykle powiązany z rolami, grupami roboczymi itp.
- ▶ Cechy
 - ▶ dodatkowy, nieobowiązkowy środek klasyfikacji
 - ▶ każdy obiekt może należeć do wielu perspektyw
 - ▶ perspektywy są określone w ramach obszaru treści
 - ▶ perspektywy mogą tworzyć hierarchie
 - ▶ na stronach perspektywy mogą być przedstawione w postaci ikon
- ▶ Wykorzystanie
 - ▶ do wyszukiwania
 - ▶ do użycia jako portlet
 - ▶ do wyświetlania w ramach folderu

Wyszukiwanie

- ▶ Rodzaje wyszukiwań
 - ▶ wyszukiwanie w obszarze treści: z użyciem kategorii, perspektyw, atrybutów
 - ▶ wyszukiwanie obiektu w portalu

- ▶ Zapamiętane wyszukiwania
 - ▶ wynik wyszukiwania można zapamiętać nadając mu nazwę
 - ▶ specjalny portlet pokazuje zapamiętane wyszukiwania

Aplikacje

Aplikacje

- ▶ Aplikacje własne Oracle Portal – przyłączone do konkretnego schematu w b.d.
- ▶ Aplikacje zewnętrzne

Komponenty aplikacji

- ▶ Formularze, raporty, wykresy, kalendarze
- ▶ Menu, hierarchie, *frame drivers* – strony przełączane listą rozwijaną
- ▶ Strony dynamiczne
- ▶ Listy wartości (LOV), łączniki (*links*)

Łączniki

- ▶ Umożliwiają wywoływanie komponentu z innego komponentu
- ▶ Pozwalają przekazywać parametry: użytkownika, systemowe (sterujące działaniem danego komponentu), z wartościami domyślnymi