

Narzędzie HTML DB do szybkiego tworzenia aplikacji internetowych – omówienie oraz ocena praktycznej przydatności

Artur Gramacki, Jarosław Gramacki

*Uniwersytet Zielonogórski
Instytut Informatyki i Elektroniki
ul. Podgórna 50, 65-246, Zielona Góra
e-mail: a.gramacki@iie.uz.zgora.pl, j.gramacki@iie.uz.zgora.pl*

Abstrakt

Celem niniejszego referatu jest bliższe przedstawienie czytelnikowi pakietu HTML DB oraz krytyczne spojrzenie na jego praktyczną przydatność i funkcjonalność. W pracy wskazane zostaną najbardziej obiecujące obszary jego zastosowania. Pakiet HTML DB został wprowadzony wraz z najnowszą wersją bazy danych Oracle 10g. Jest to narzędzie do szybkiego tworzenia aplikacji w środowisku internetowym (ang. *Rapid Web Application Development Tool*). HTML DB jest przeznaczony dla tych użytkowników, którzy muszą szybko przygotować w miarę funkcjonalną, ale dość prostą aplikację. Nie jest przy tym wymagana dogłębna znajomość żadnych innych technik programowania aplikacji dla baz danych Oracle. Całość procesu tworzenia aplikacji odbywa się w przeglądarce internetowej z pomocą narzędzia będącego bardzo rozbudowanym kreatorem.

Informacja o autorach

dr inż. Artur Gramacki – pracuje w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego na stanowisku adiunkta. Jego zainteresowania koncentrują się wokół szeroko rozumianych zagadnień związanych z bazami danych, w szczególności firmy Oracle. Oprócz prowadzenia zajęć dydaktycznych stara się wykorzystywać swoją wiedzę uczestnicząc w różnych projektach informatycznych z tego zakresu. Brał udział w trzech projektach, których celem było przygotowanie systemów wspomagających działalność Uniwersytetu Zielonogórskiego.

dr inż. Jarosław Gramacki – jest pracownikiem naukowym w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Zajmuje się projektowaniem, wykonywaniem oraz wdrażaniem aplikacji bazodanowych usprawniających szeroko rozumiane zarządzanie Uczelnią. Od wielu lat prowadzi również zajęcia dydaktyczne dotyczące projektowania baz danych, działania i administrowania systemami zarządzania bazami danych oraz wykorzystania technologii Oracle w budowie aplikacji użytkowych.

1. Wstęp

Aby realnie wykorzystywać system Oracle niezbędne jest posiadanie narzędzi umożliwiających tworzenie aplikacji użytkowych. Można wręcz zaryzykować stwierdzenie, że bez nich sens posiadania bazy danych jest co najmniej dyskusyjny. Firma Oracle w zasadzie od początku swojego istnienia dostarcza tego typu narzędzia. Najnowszą propozycją w tej grupie produktów jest pakiet HTML DB [1].

Jeszcze do niedawna w świecie baz danych niekwestionowanym liderem były rozwiązania umożliwiające tworzenie aplikacji typu klient-serwer. W tym segmencie propozycją firmy Oracle był pakiet *Developer* i dwa jego kluczowe narzędzia *Forms* oraz *Reports*. Pomimo tego, że pakiet ten jest ciągle intensywnie rozwijany, celowo użyto tutaj czasu przeszłego (był), gdyż najnowsze wcielenie pakietu *Developer (Oracle Developer Suite 10g)* praktycznie likwiduje możliwość tworzenia aplikacji w architekturze klient-serwer na rzecz aplikacji internetowych w architekturze trójwarstwowej.

Ostatnie lata to czas gwałtownego rozwoju aplikacji internetowych i związanych z nimi narzędzi. Aplikacje typu klient-serwer powoli zaczynają ustępować im pola (można się zastanowić, czy jest to rzeczywista wola użytkowników, czy też działanie celowo wymuszane przez twórców narzędzi programistycznych). W tym z kolei segmencie jak na razie niezachwianą pozycję posiada język Java i wszelkie związane z nim technologie (podstawowe to: serwlety, strony JSP oraz tzw. ziarna Javy czyli *JavaBeans*). Bardzo popularny wśród użytkowników systemu Oracle jest również język PL/SQL, który także umożliwia efektywne tworzenie aplikacji internetowych (z wykorzystaniem modułu *mod_plsql* serwera Oracle HTTP *Apache*).

Tworzenie aplikacji internetowych w językach Java oraz PL/SQL nie jest jednak zbyt łatwym przedsięwzięciem. Oba rozwiązania wymagają poznania dość złożonych narzędzi i technik programistycznych. O ile opanowanie języka PL/SQL jest stosunkowo proste, to tego samego nie można już powiedzieć o języku Java. Z jednej strony język ten powszechnie uważany jest za zdecydowanie łatwiejszy do opanowania niż języki C/C++ (w których również można tworzyć aplikacje dla bazy danych Oracle). Z drugiej natomiast strony w przypadku wykorzystania go do tworzenia aplikacji internetowych dla baz danych, zmuszeni jesteśmy do poznania wielu dodatkowych technik i standardów, które z pewnością nie należą do najłatwiejszych. Aby więc tworzyć aplikacje internetowe dla baz danych Oracle, wymagane jest spore doświadczenie i umiejętności programistyczne. Co więcej, twórca takich aplikacji musi mieć wiedzę nie tylko na temat *stricte* języków Java i/lub PL/SQL, ale również musi, przynajmniej w stopniu średnio zaawansowanym, rozumieć działanie serwera bazy danych Oracle. Techniki te są więc raczej predysponowane do tworzenia systemów o większych rozmiarach i dużym stopniu złożoności, przez zespoły doświadczonych programistów.

Z drugiej jednak strony często jest tak, że tworzone aplikacje są stosunkowo niewielkie i nie mają one znaczenia strategicznego w firmie a dodatkowo otrzymany przez zleceniodawcę czas na ich wykonanie jest bardzo ograniczony. W takich właśnie sytuacjach bardzo ciekawą propozycją wydaje się być nowe narzędzie firmy Oracle pod nazwą HTML DB. Celem niniejszego artykułu jest bliższa jego prezentacja.

Pakiet HTML DB pojawił się wraz nową bazą danych Oracle *10g*. Jest to kompletne środowisko do budowy aplikacji internetowych. Narzędzie HTML DB obsługiwane jest wyłącznie z poziomu przeglądarki internetowej i tylko z tego poziomu użytkownik wykonuje wszelkie czynności konfiguracyjne, projektuje i implementuje aplikację i w końcu udostępnia ją użytkownikom końcowym.

Aby efektywnie używać tego narzędzia nie jest konieczne zbyt wielkie doświadczenie programistyczne – choć gwoli prawdy jego brak może skutecznie odstraszyć przyszłego użytkownika. Pakiet HTML DB nie jest wcale taki prosty, jak by to mogło się z pozoru wydawać.

2. Instalacja i konfiguracja

2.1. Instalacja

Aby zainstalować pakiet HTML DB należy w pierwszej kolejności pobrać ze stron OTN (otn.oracle.com) płytę instalacyjną o nazwie *Oracle 10g Companion CD*. Instalację prowadzimy z pomocą znanego z innych produktów Oracle-a uniwersalnego instalatora (ang. *Universal Installer*).

Pakiet HTML DB z technicznego punktu widzenia stanowi bardzo rozbudowaną aplikację internetową napisaną z wykorzystaniem bramki sieciowej PL/SQL. Wymagany jest więc serwer WWW (*Apache*) z prawidłowo skonfigurowanym modulem *mod_plsql*. Serwer taki jest automatycznie instalowany i konfigurowany w czasie instalowania pakietu HTML DB.

Instalacja [4, 5] musi być wykonana w nowym katalogu domowym (ang. *Home directory*). Przed rozpoczęciem instalacji musimy upewnić się, że posiadamy dostęp (oraz znamy hasło użytkownika *SYS*) do bazy danych w wersji *9i* (co najmniej 9.2.0.3) lub bazy *10g*. Dostęp do bazy jest konieczny, gdyż pakiet HTML DB nie stanowi niezależnego produktu – jest to *de facto* zbiór obiektów bazy Oracle, które tworzą całą wymaganą „infrastrukturę”. W jego skład wchodzi w szczególności:

- 147 tabel relacyjnych,
- 132 pakiety (w każdym od kilku do ponad 30 procedur i funkcji),
- 160 wyzwalaczy.

Jak widać jest to całkiem spora ilość elementów, co czyni pakiet HTML DB dość złożonym produktem.

Instalacja trwa dość długo – zwłaszcza etap konfiguracji, gdy do bazy danych musi zostać załadowana bardzo duża liczba programów PL/SQL. Automatycznie instalowany jest również serwer HTTP *Apache* (nie ma możliwości rezygnacji z jego instalacji). W przypadku, gdy pakiet HTML DB instalowany jest na bazie *10g*, proces konfiguracji deskryptorów dostępowych do bazy (ang. *Database Access Descriptor*; w skrócie *DAD*) wykonywany jest automatycznie. W przypadku bazy *9i* konfigurację taką trzeba niestety wykonać ręcznie.

Po wykonaniu powyższych czynności możemy przetestować poprawność instalacji i konfiguracji wpisując w przeglądarce WWW adres strony:

```
http://hostname:7777/htmldb/htmldb_admin.
```

W pierwszym kroku musimy zalogować się na konto administratora systemu HTML DB. Standardowo nazwa tego konta brzmi *ADMIN* a jego hasło jest takie, jakie podaliśmy na początku instalacji. Jako *hostname* podajemy nazwę komputera, gdzie instalowany był pakiet HTML DB. Gdy przed rozpoczęciem instalacji port numer 7777 był już zajęty, instalator użyje innego wolnego numeru. Informacja o tym zostanie zapisana w pliku `<HTMLDB_HOME>/install/portlist.ini`. W efekcie na ekranie powinniśmy zobaczyć stronę pokazaną na Rysunku 1. W tym momencie możemy rozpocząć pracę z pakietem.

Rys. 1. Strona domowa modułu administracyjnego HTML DB

Gdy pokazana na Rysunku 1 strona nie pojawia się, możemy spróbować wykonać szybki test poprawności instalacji pakietu [7]. W pierwszej kolejności powinniśmy odblokować (ang. *unlock*) konto użytkownika *flows_010500* (w czasie normalnej pracy pakietu HTML DB konto to powinno być zablokowane i tak też czyni program instalacyjny). Następnie powinniśmy zalogować się z poziomu programu *SQL*Plus* na to konto. Hasło jest takie samo jak to, które zostało określone na początku instalacji pakietu. Następnie możemy wykonać polecenia wymienione na Rysunku 2. Gdy nie są sygnalizowane żadne błędy *ORA-XXXXX*, możemy założyć, że część bazodanowa pakietu pracuje poprawnie. Gdy aplikacja nadal nie działa, błędów należy szukać w niewłaściwej konfiguracji deskryptora dostępowego *DAD*.

```
D:\oracle\ora10g\bin>sqlplus /nolog
SQL*Plus: Release 10.1.0.2.0 - Production on Ir Wrz 15 13:02:57 2004
Copyright (c) 1982, 2004, Oracle. All rights reserved.
SQL> connect flows_010500
Enter password:
Połączono.
SQL>
SQL> declare nm owa.vc_arr;
2   vl owa.vc_arr;
3   begin
4     nm(1) := 'WEB_AUTHENT_PREFIX';
5     vl(1) := 'WEB$';
6     vl(2) := 'no cookie';
7     owa.init_cgi_env(1,nm,vl);
8   end;
9   /

Procedura PL/SQL została zakończona pomyślnie.
```

```
SQL>
SQL> set serveroutput on size 100000
SQL>
SQL> exec f (p=>'4550:1');

PL/SQL procedure successfully completed.

SQL>
SQL> exec owa_util.showpage
-- Otrzymany strumień musi "przypominać" kod HTML

Content-type: text/html
Content-length: 6005
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN">
<html>
<head>
...

PL/SQL procedure successfully completed.

SQL>
```

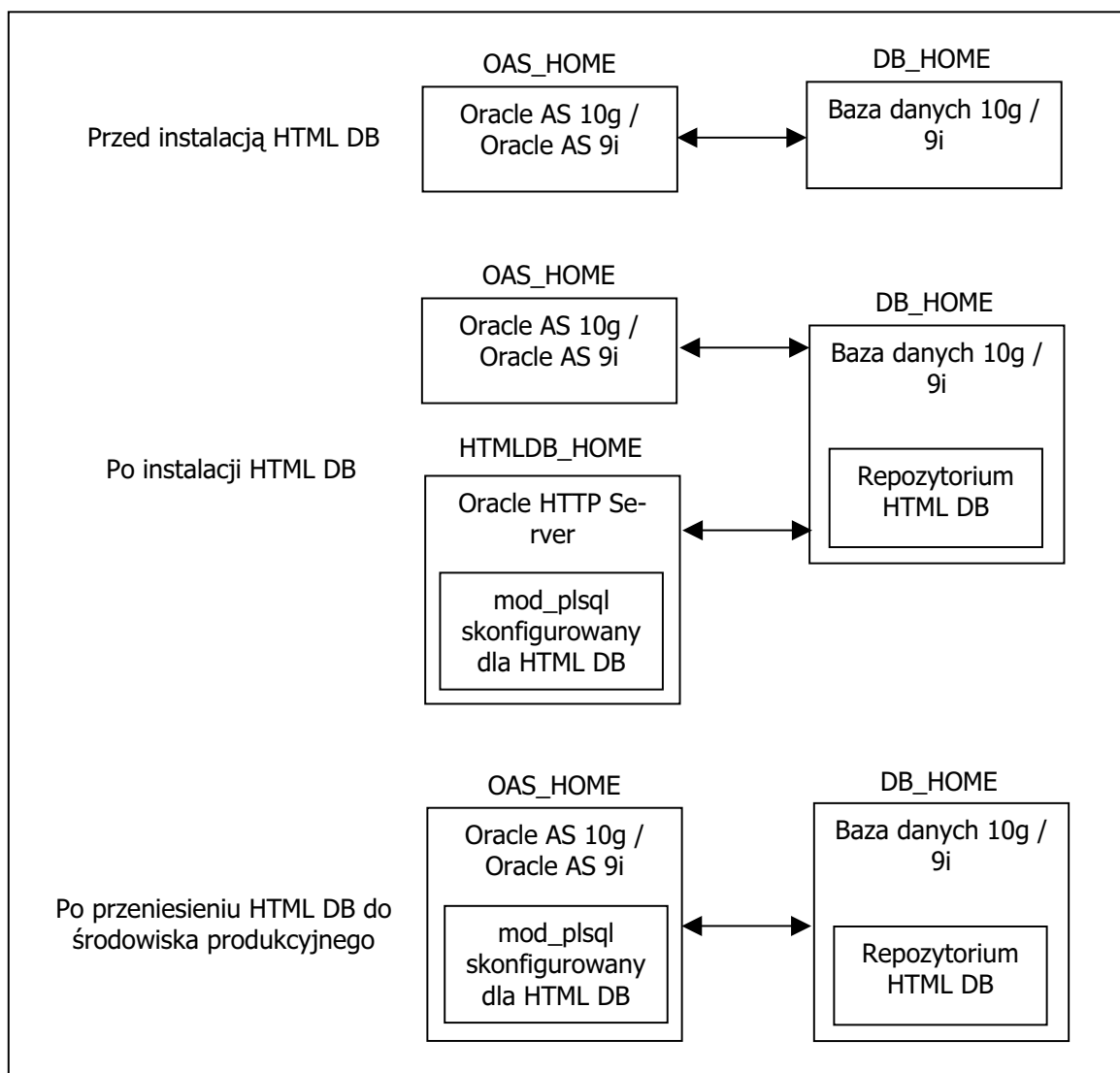
Rys. 2. Szybki test poprawności instalacji części bazodanowej pakietu HTML DB

W efekcie opisanego powyżej (w bardzo skrótovej formie) procesu instalacji pakietu HTML DB, na dysku serwera pojawi się odpowiednio skonfigurowany do pracy z HTML DB serwer HTTP *Apache*. Do celów testowych taka konfiguracja jest jak najbardziej odpowiednia, jednak w środowisku produkcyjnym wolelibyśmy raczej, aby HTML BD mógł pracować z serwerem HTTP będącym częścią serwera aplikacyjnego *10g* lub *9i* (tylko z tymi wersjami pakiet HTML DB współpracuje prawidłowo). Po co mianowicie mnożyć ponad miarę ilość zainstalowanych serwerów HTTP – ten, który istnieje w produkcyjnym serwerze aplikacyjnym najprawdopodobniej z powodzeniem może obsłużyć jeszcze jedną aplikację.

W takim przypadku należy dokonać odpowiedniej konfiguracji (ręcznie, gdyż automatyczna procedura na dzień dzisiejszy – wrzesień 2004 – nie istnieje). Polega to najogólniej mówiąc na tym, że kopiujemy wymagane pliki z katalogu domowego HTML DB do katalogu domowego serwera aplikacyjnego a następnie modyfikujemy wymagane pliki konfiguracyjne tegoż serwera. Po upewnieniu się, że HTML DB działa prawidłowo w środowisku produkcyjnym, możemy już za pomocą uniwersalnego instalatora Oracle, usunąć wykonaną na samym początku instalację HTML DB. Na Rysunku 3 pokazano poszczególne etapy instalacji.

Opisany powyżej proces migrowania do środowiska produkcyjnego, chociaż nie jest zbyt skomplikowany i jest dobrze udokumentowany [2, 3], to mimo wszystko wydaje się nieco dziwny. Być może w przyszłych wersjach pakietu HTML DB firma Oracle zapewni bardziej elegancki sposób instalowania produktu bezpośrednio w środowisku produkcyjnym, bez potrzeby instalowania tymczasowej wersji „rozruchowej”.

Bliższe przyjrzenie się zawartości katalogów pakietu HTML DB doprowadziło do „odkrycia” katalogu <HTML_DB>/marvel/doc i pliku *installationguide.html*, gdzie znajduje się dość szczegółowy opis ręcznej instalacji wymaganych programów PL/SQL. Wydaje się więc, że płytę instalacyjną *Oracle 10g Companion CD* w obecnej wersji należy traktować jako przejściową.



Rys. 3. Poszczególne etapy instalacji pakietu HTML DB oraz migracji do środowiska produkcyjnego

2.2. Deinstalacja

Aby usunąć z bazy danych obiekty tworzące HTML DB należy, po połączeniu się na użytkownika *SYS* lub *SYSTEM*, wykonać następujące czynności [4]:

```
SQL> ALTER SESSION SET current_schema = FLOWS_010500;
SQL> EXEC wwv_flow_upgrade.drop_public_synonyms;
SQL> ALTER SESSION SET current_schema = SYSTEM;
SQL> DROP USER flows_010500 CASCADE;
SQL> DROP USER flows_files CASCADE;
SQL> DROP USER htmldb_public_user CASCADE;
```

Analiza powyższych poleceń dość dobrze obrazuje czym w strukturze bazy danych jest pakiet HTML DB. Należy podejrzewać, że nazwa użytkownika *flows_010500* może ulec zmianie wraz z pojawianiem się kolejnych wersji produktu.

Uwaga: użytkownik *flows_010500* ma bardzo duże uprawnienia, zbliżone do użytkownika *SYS*. Można zadać sobie pytanie, czy rzeczywiście do pracy pakietu HTML DB potrzeba jest aż tyle uprawnień (np. *ALTER DATABASE*).

3. Dla kogo stworzono pakiet HTML DB

HTML DB to najnowsza propozycja firmy Oracle skierowana do twórców aplikacji internetowych. Należy jednak wyraźnie doprecyzować o jakich twórców aplikacji tutaj chodzi. Na pewno nie są to doświadczeni developerzy, piszący swoje aplikacje w takich językach jak Java, PHP czy PL/SQL (lub inne „prawdziwe” języki programowania). Oni z pewnością będą postrzegali pakiet HTML DB jako niezbyt poważny produkt do mało ważnych i raczej prostych zastosowań.

I tak jest w istocie. Twórcy pakietu HTML DB tworzyli go od samego początku dla osób, które muszą w miarę szybko (w ciągu kilku dni a nie tygodni, czy miesięcy) stworzyć dość ładnie wyglądającą aplikację o podstawowej funkcjonalności, nie posiadając przy tym wiedzy i doświadczenia w tworzeniu aplikacji internetowych. Aplikację w HTML DB tworzy się w zasadzie niemal całkowicie metodą nazywaną popularnie „myszkologiczną”. Oznacza to, że twórca aplikacji nie musi (przynajmniej teoretycznie) znać żadnego „normalnego” języka programowania, aby poradzić sobie z budową aplikacji. Jeżeli dodatkowo zadowolony się predefiniowanymi szablonami graficznymi i nie będzie chciał zbytnio zmieniać założonego przez twórców HTML DB poziomu złożoności aplikacji, jest w stanie stworzyć je w iście ekspresowym tempie.

Niestety, to co jest zaletą dla niewprawnego programisty, będzie z pewnością dużą wadą dla kogoś bardziej doświadczonego. Taka osoba z pewnością niezbyt chętnie sięgnie po to narzędzie. Ilości okienek, które trzeba przejść, aby zbudować nawet najprostszą aplikację wystawia na ciężką próbę cierpliwość „prawdziwego” programisty.

Wydaje się, że potencjalnymi użytkownikami pakietu HTML DB mogą być również administratorzy baz danych Oracle, gdyż jak pokazują przykłady wielu firm (czy tylko polskich?), administrator bardzo często zmuszony jest do tworzenia, mniej lub bardziej złożonych, aplikacji bazodanowych. Zwykle tworzy je on niejako z doskoku, gdyż jego podstawowym obowiązkiem jest oczywiście dbanie o bezpieczną pracę serwerów bazodanowych. Można więc spodziewać się, że pakiet HTML DB zostanie w tej grupie użytkowników przyjęty jeżeli nie entuzjastycznie to na pewno z pewnego rodzaju przychylnym zainteresowaniem.

4. Jak działa pakiet HTML DB

4.1. Repozytorium

HTML DB składa się z repozytorium instalowanego w schemacie użytkownika o nazwie *flows_010500*. Każda aplikacja użytkownika zapisywana jest w postaci wierszy w odpowiednich tabelach relacyjnych tego repozytorium. Można więc mówić o swego rodzaju metadanych. Żądanie uruchomienia aplikacji HTML DB powoduje wykonanie, za pośrednictwem modułu *mod_plsql*, odpowiednich procedur PL/SQL, które na podstawie informacji zapisanych w repozytorium wyrysowują na ekranie żadaną stronę.

Widzimy więc, że kodem źródłowym aplikacji HTML DB są po prostu dane (metadane) zapisane w repozytorium. Taki sposób tworzenia w trybie *on-line* poszczególnych stron aplikacji powoduje, że całość działa nadzwyczaj szybko i stabilnie (nie mamy tu oczywiście na myśli niestabilności wynikającej z ew. błędów w oprogramowaniu pakietu HTML DB; mamy na myśli wyłącznie stabilność gwarantowaną przez moduł *mod_plsql* oraz serwera *Apache*).

4.2. Przestrzenie robocze i użytkownicy

Aby rozpocząć tworzenie aplikacji w HTML DB musimy zalogować się na utworzone wcześniej konto. Konta tworzone są w ramach tzw. przestrzeni roboczych (ang. *workspace*). Przestrzenie robocze może tworzyć administrator pakietu HTML DB (patrz podrozdział 2.1). W każdej przestrzeni roboczej musi istnieć przynajmniej jeden użytkownik, który jest administratorem tej przestrzeni. Nazwa tego użytkownika oraz hasło ustalane są w czasie tworzenia przestrzeni.

Uwaga: administrator przestrzeni roboczej to nie to samo co administrator pakietu HTML DB. Jego uprawnienia ograniczają się tylko do jednej wybranej przestrzeni, podczas gdy administrator HTML DB zarządza pakietem jako całością.

Wszelkie czynności administracyjne wykonywane są z poziomu strony (i jej podstron) pokazanej na Rysunku 1. Każda przestrzeń robocza tworzona jest w ramach wybranego schematu użytkownika. Można użyć schematu już istniejącego lub też zlecić utworzenie odpowiedniego schematu w czasie tworzenia przestrzeni roboczej. W jednym schemacie użytkownika może być utworzonych wiele przestrzeni roboczych.

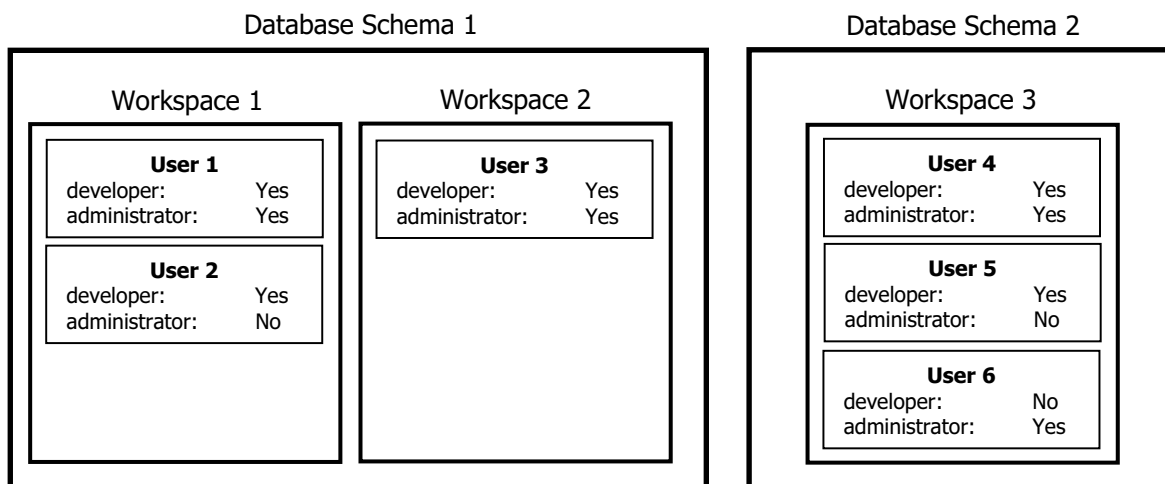
Mając utworzone przestrzenie robocze możemy następnie utworzyć konta dla poszczególnych użytkowników i/lub developerów. Każde konto może mieć przypisane dwa rodzaje uprawnień *Administrator* i/lub *Developer*. Gdy nie ma przypisanego żadnego z nich, staje się on użytkownikiem, który może tylko uruchamiać aplikacje stworzone w HTML DB. Uprawnienie *Administrator* umożliwia zarządzanie bieżącą przestrzenią roboczą, w tym w szczególności:

- zarządzanie użytkownikami,
- monitorowanie pracy użytkowników,
- wgląd do logów systemowych,
- zarządzanie stanem sesji,
- tworzenie raportów podsumowujących działalność użytkowników.

Na Rysunku 4 pokazano wzajemne zależności pomiędzy schematami bazy danych, przestrzeniami roboczymi oraz kontami użytkowników.

Aby zalogować się na konto użytkownika przestrzeni roboczej wpisujemy w przeglądarce WWW adres strony:

`http://hostname:7777/htmldb/htmldb_login`



Rys. 4. Relacje pomiędzy schematem bazy danych, przestrzenią roboczą oraz użytkownikiem

Na Rysunku 5 pokazano ekran startowy po zalogowaniu się na konto administratora przestrzeni roboczej. Po zalogowaniu się na konto bez uprawnień *Administrator*, nie będzie widoczna ikona *Administration*.

4.3. Komponenty HTML DB

W obecnej wersji pakiet HTML DB składa się z 4 komponentów, służących do wykonywania określonych zadań w czasie administrowania pakietem oraz tworzenia w nim aplikacji. Są one skrótowo omówione poniżej.

4.3.1. SQL Workshop

Komponent ten umożliwia interakcyjną pracę z bazą danych. Umożliwia wydawanie poleceń SQL oraz PL/SQL, wykonywanie skryptów, wygodne oglądanie słownika bazy danych, tworzenie nowych obiektów oraz generowanie poleceń DDL dla istniejących w bieżącym schemacie obiektów. Ogólnie można powiedzieć, że funkcjonalnie jest to połączenie możliwości oferowanych przez program *SQL*Plus* oraz takie narzędzie graficzne jak np. program *Toad* firmy *PowerQuest* (ten ostatni ma oczywiście dużo większe możliwości niż *SQL Workshop*).

4.3.2. Data Workshop

Komponent ten umożliwia importowanie oraz eksportowanie danych do/z bazy danych. Źródłem mogą być pliki tekstowe, pliki XML oraz dane pobrane z arkuszy roboczych, np. z programu Excel. Całość działa bardzo sprawnie i szybko.



Rys. 5. Strona domowa przestrzeni roboczej

4.3.3. Administration

Komponent ten umożliwia wykonywanie czynności administracyjnych dotyczących bieżącej przestrzeni roboczej. Czynności możliwe do wykonania skrótowo omówiono w podrozdziale 4.2.

4.3.4. Application Builder

Jest to najważniejszy i najbardziej złożony komponent wchodzący w skład pakietu HTML DB. Z jego pomocą tworzymy aplikacje internetowe. Jest on w istocie bardzo rozbudowanym kreatorem, z dużą ilością predefiniowanych elementów interfejsu użytkownika (ang. *widget*, skrót od *window gadget*).

Aby zorientować się w możliwościach pakietu najprościej jest uruchomić domyślnie instalowaną w każdej nowotworzonej przestrzeni roboczej aplikację. Nosi ona nazwę *Sample Application* i ma przypisany numer 100. Uruchamiamy ją w ten sposób, że po poprawnym zalogowaniu się do przestrzeni roboczej wybieramy komponent *Application Builder* a następnie wskazujemy z listy rozwijalnej aplikację demonstracyjną (na początku na liście jest tylko ta jedna aplikacja). Następnie klikamy na ikonę *Run* – gdy nie pojawi się żaden błąd powinniśmy zobaczyć stronę startową aplikacji pokazaną na Rysunku 6. Bardziej szczegółowe omówienie komponentu *Application Builder* zamieszczono w kolejnym rozdziale.

Pakiet HTML DB wyposażono domyślnie w cztery aplikacje demonstracyjne (jedną z nich jest opisana powyżej *Sample Application*). W dowolnym momencie każda z nich może zostać zainstalowana od nowa, przeinstalowana lub usunięta z przestrzeni roboczej. Aplikacje te dostępne są po wybraniu linku *Review Demonstrations* dostępnego z poziomu strony głównej *Application Builder*-a (patrz Rysunek 7).

The screenshot shows the Oracle Sample Application home page. At the top left is the Oracle logo. To the right are links for 'Help Print Logout'. Below the logo is a navigation bar with four tabs: 'Home', 'Orders', 'Products', and 'Customers'. A label 'zakładki' points to these tabs. On the left side, there is a 'Customer Search' section with a search box and a 'Go' button. Below it is a 'Tasks' list with links: 'Enter New Order', 'Enter New Customer', and 'About this Application'. A label 'przycisk' points to the 'Go' button. The main content area is divided into three sections: 'Top Orders', 'Top Customers', and 'Top Products'. 'Top Orders' is a table with columns: Order #, Order Date, Customer Name, Order Total. 'Top Customers' is a table with columns: Customer Name, Total Revenue. 'Top Products' is a horizontal bar chart showing product names and their revenue. A label 'regiony' points to the 'Top Customers' table. At the bottom, there is a 'pasek zadań' (taskbar) containing links: 'Edit Application', 'Edit Page', 'NewSession', 'Debug', 'Show', 'Edit Links', and 'Links'.

Rys. 6. Strona domowa przykładowej aplikacji demonstracyjnej

5. Komponent Application Builder

5.1. Uruchamianie aplikacji

Każda aplikacja HTML DB jest zbiorem stron (ang. *pages*), które są odpowiednikiem okienek w systemie Windows. Strona jest podstawową jednostką projektową w HTML DB. Przykładowa aplikacja pokazana na Rysunku 6 składa się z 16 stron. Ich nazwy i numery widoczne są na stronie domowej komponentu *Application Builder*. Pokazano ją na Rysunku 7.

Na każdej stronie możemy definiować praktycznie wszystkie elementy GUI znane z aplikacji okienkowych (np. przyciski, listy rozwijalne, pola edycyjne, ikony itd.). Poszczególne elementy możemy grupować w logiczne bloki zwane regionami (ang. *regions*). Dodatkowo możemy tworzyć zakładki, ułatwiające nawigowanie po aplikacji. Niektóre z tych elementów zaznaczono na Rysunku 6. Na dole każdej strony znajduje się pasek zadań, który na etapie tworzenia aplikacji daje szybki dostęp do najczęściej wykonywanych czynności. W aplikacji udostępnianej końcowemu użytkownikowi pasek ten nie powinien być oczywiście widoczny.



Rys. 7. Strona domowa komponentu Application Builder

Z technicznego punktu widzenia nawigacja między poszczególnymi stronami tworzącymi aplikację odbywa się według jednolitego schematu. Każda aplikacja posiada swój unikalny w ramach przestrzeni roboczej numer lub alfanumeryczny alias (ang. *application ID*). Także każda strona posiada swój unikalny numer lub alfanumeryczny alias w ramach aplikacji (ang. *page ID*). Po uruchomieniu aplikacji motor HTML DB nadaje jej unikalny numer sesyjny (ang. *session number*), dzięki któremu system jest w stanie dokładnie zidentyfikować poszczególne instancje aplikacji (każda aplikacja może zostać uruchomiona przez wielu użytkowników równocześnie). Podając te parametry w adresie URL dostajemy się do interesujących nas stron. Przykładowo wpis:

```
http://localhost:7777/pls/htmldb/f?p=103:1:5203597145955937078
```

odczytamy następująco:

- wirtualny adres pakietu HTML DB: `http://localhost:7777/pls/htmldb`,
- unikalny numer aplikacji: 103,
- numer żądanej strony: 1,
- unikalny numer sesyjny: 5203597145955937078.

Powyższy przykład nie zawiera wszystkich możliwych do wyspecyfikowania parametrów uruchomieniowych. Pełna składnia jest następująca (dokładny opis można znaleźć w dokumentacji [6]):

```
f?p=App:Page:Session:Request:Debug:ClearCache:itemNames:itemValues:PrinterF  
riendly
```

5.2. Eksport oraz import aplikacji HTML DB

Ponieważ aplikacja HTML DB przechowywana jest w repozytorium w postaci metadanych, więc aby efektywnie przenosić aplikacje między różnymi serwerami musi być stworzony odpowiedni mechanizm eksportu i importu aplikacji.

Wyeksportowanie aplikacji HTML DB oznacza w praktyce automatyczne wygenerowanie skryptu *SQL*Plus*, który zawiera wszystkie niezbędne polecenia SQL oraz PL/SQL pozwalające na zbudowanie aplikacji (czyli zaimportowanie jej) w docelowej bazie danych z zainstalowanym pakietem HTML DB (miejmy nadzieję, że w przyszłych wersjach produktu zostanie zachowana wsteczna kompatybilność !).

Oprócz poszczególnych aplikacji eksportować można również definicje przestrzeni roboczych. W ten sposób możliwe staje się przenoszenie całego środowiska developerskiego a w szczególności użytkowników, grup użytkowników oraz ich przywilejów. Podobnie jak przy eksportowaniu aplikacji w wyniku otrzymujemy skrypt *SQL*Plus*-a.

5.3. Metody budowania aplikacji

Budowanie aplikacji HTML DB może odbywać się na kilka różnych sposobów. Po wybraniu ze strony domowej komponentu *Application Builder* (patrz Rysunek 7) ikony *Create Application* dostajemy możliwość wyboru jednej z trzech metod tworzenia aplikacji. Są to:

- tworzenie aplikacji całkowicie ręcznie (ang. *From Scratch*). Ta metoda jest najbardziej czasochłonna. Wymaga poznania bardzo dokładnie pakietu HTML DB. Daje jednak stosunkowo duże możliwości kreowania formy graficznej przyszłej aplikacji,
- tworzenie aplikacji na bazie aplikacji już istniejącej (ang. *Based on an Existing Application*). Nazwa jest nieco myląca, gdyż chodzi tutaj o utworzenie nowej aplikacji zawierającej szablony (ang. *templates*) istniejące w kopiowanej aplikacji. Dzięki temu nowotworzona aplikacja może mieć identyczną szatę graficzną jak aplikacja wzorcowa,
- tworzenie aplikacji na bazie istniejących tabel (ang. *Based on Existing Tables*). Tworzy niemal całkowicie automatycznie prostą aplikację zawierającą trzy elementy: formularze do edycji danych, formularze do tworzenia nowych rekordów oraz proste wykresy (ang. *charts*). Przykładowo dla standardowych tabel EMP oraz DEPT otrzymujemy aplikację pokazaną na Rysunku 8. Uwaga: do oglądania wykresów wymagany jest program *Adobe SVG Viewer 3* (dostępny bezpłatnie ze strony: <http://www.adobe.com/svg/viewer/install/main.html>).

Można również korzystać z wbudowanych wzorników (ang. *wizard*). Pozwalają one w ekspresowym tempie tworzyć strony z formatkami do wprowadzania/edycji danych oraz proste raporty. Strony te mogą następnie być użyte jako gotowe komponenty w budowanej aplikacji.



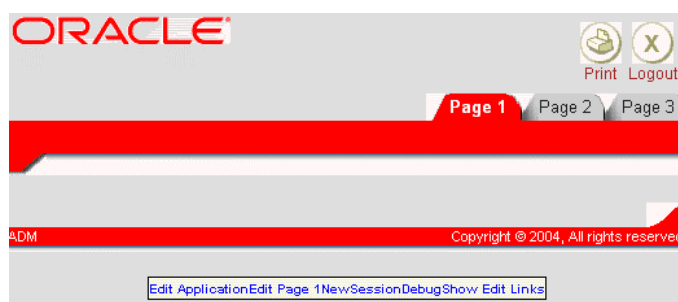
Rys. 8. Strona startowa aplikacji utworzonej metodą *Based on Existing Tables*

5.4. Przykład budowy aplikacji

Poniżej zostanie w skrótovej formie proces budowania aplikacji metodą *From Scratch*, czyli tą, która mimo iż jest najbardziej pracochłonna, daje największe możliwości.

Opisanie szczegółowo całego procesu nie jest możliwe ze względu na ograniczoną objętość artykułu. Opis zawężono więc do pokazania najważniejszych jego etapów. Wszystkie praktycznie czynności projektowe wykonuje się przy pomocy myszki.

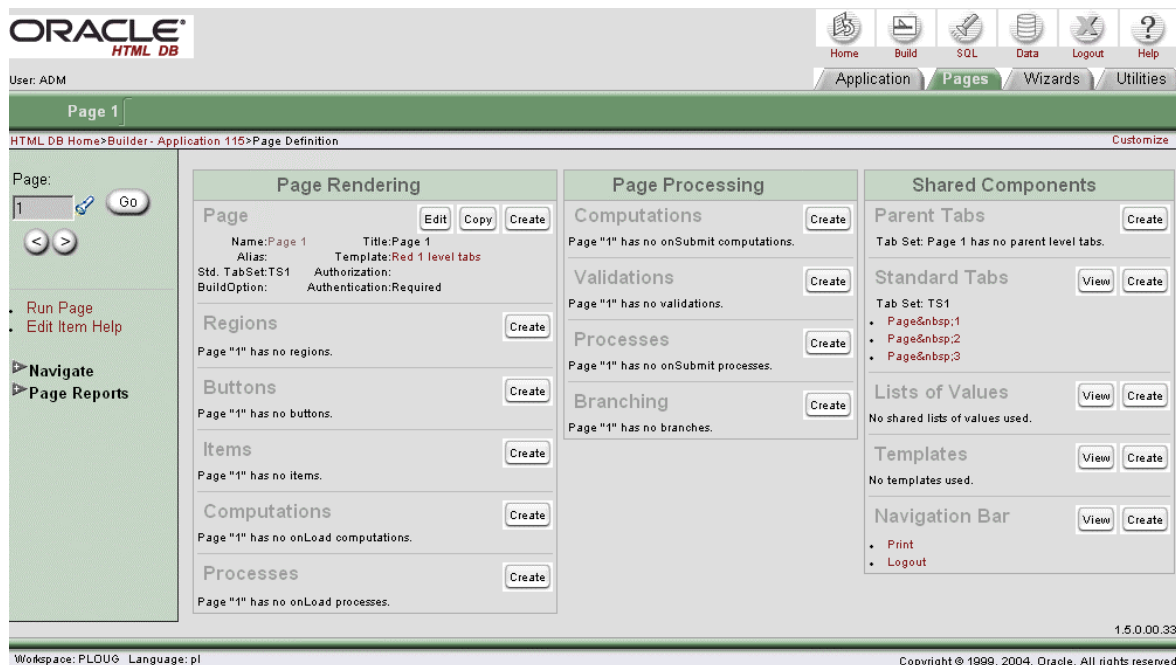
W pierwszym kroku należy zdefiniować nazwę aplikacji, określić, czy aplikacja będzie wymagała autoryzacji oraz wybrać formę graficzną (szablony, na bazie których powstanie aplikacja). Po wykonanie kilku prostych czynności (komputer prowadzi nas za rękę – nie sposób się pomylić) otrzymujemy aplikację (w zasadzie dopiero jej szablony), która po uruchomieniu wygląda tak jak to pokazano na Rysunku 9.



Rys. 9. Szablon pierwszej aplikacji HTML DB

W kolejnym kroku, na poszczególnych stronach tworzymy regiony i całą związaną z nimi logikę aplikacji. Ten etap jest bardzo (!) czasochłonny. Każdy najdrobniejszy element aplikacji definiowany jest z poziomu jednego z bardzo wielu formularzy HTML. Na każdym z tych formularzy jest znaczna liczba elementów interfejsu użytkownika. Bardzo szczegółowy ich opis można znaleźć w [6], jednak autorzy artykułu przestrzegają, że dokumentacja ta, choć bardzo kompletna, jest bardzo mało komunikatywna (co nawiasem mówiąc jest cechą dużej części oficjalnej dokumentacji tworzonej przez firmę Oracle).

Edycję każdej ze stron rozpoczynamy od ekranu pokazanego na Rysunku 10. Są na nim trzy główne panele: do tworzenia i edycji elementów graficznych na stronie (ang. *Page Rendering*), do tworzenia logiki aplikacji (ang. *Page Processing*) oraz do tworzenia elementów współdzielonych (ang. *Shared Components*).



Rys. 10. Strona startowa do budowania aplikacji

6. Podsumowanie

6.1. Wrażenia ogólne

Autorzy pakietu HTML DB postawili sobie bardzo ambitne zadanie. Postanowili mianowicie stworzyć narzędzie, z pomocą którego osoby bardzo niedoświadczone w dziedzinie programowania, będą w stanie tworzyć mimo wszystko dość złożone aplikacje. Niestety okazuje się, że nie jest to zadanie łatwe. Aplikacje HTML DB można co prawda tworzyć bez znajomości praktycznie żadnych technik i języków programowania, ale cóż z tego, gdy i tak trzeba poznać mnóstwo szczegółów pakietu. Aby efektywnie wykorzystywać to narzędzie trzeba najpierw spędzić wiele dni na wgłębianiu się w jego detale. Chociaż aplikacja HTML DB nie posiada kodu źródłowego w powszechnym rozumieniu, to metadane oraz motor HTML DB, które go w pewnym sensie zastępują, nie są wiele prostsze do ogarnięcia niż kod źródłowy w postaci klasycznego pliku ASCII. Można więc zaryzykować stwierdzenie, że dobre opanowanie pakietu HTML DB wymaga podobnych zdolności i predyspozycji, które wymagane są do opanowanie jakiegoś prostszego języka programowania (jak np. PL/SQL czy PHP). Wydaje się więc, że przedstawianie na stronach Oracle pakietu HTML DB jako coś prostego i łatwego do opanowania jest nieco przesadzone. Jest niemal pewne, że osoba nie mająca żadnego doświadczenia w programowaniu, nie poradzi sobie z obsługą HTML DB.

Gdy już w miarę dobrze poznamy pakiet HTML DB, musimy uzbroić się w potężną dawkę cierpliwości i bardzo dobrej jakości myszki. Modyfikacja praktycznie każdego elementu wymaga poruszania się po wielu, często bardzo długich, formularzach HTML. W zasadzie większość czasu

spędzamy na „dokopywaniu” się do miejsc, z poziomu których możemy zmienić interesujący nas element.

Trzeba jednak przyznać, że stworzenie aplikacji o podobnej funkcjonalności ale za pomocą klasycznych technik (Java, PL/SQL, PHP) zajęłoby na pewno więcej czasu – choć w efekcie otrzymamy łatwiej przenaszalny kod źródłowy, do którego łatwiej będzie wprowadzać nową funkcjonalność (inną niż oferowana standardowo przez pakiet HTML DB).

6.2. Stabilność i estetyka

Na uwagę zasługuje całkiem dobra stabilność pakietu HTML DB. Autorzy artykuły tylko sporadycznie spotkali się z błędnym działaniem systemu, choć gwoździ prawdy budowane przez nich aplikacje nie wykorzystywały wszystkich możliwości pakietu i być może są w nim jakieś bardziej znaczące błędy, które ujawnią się dopiero w trakcie wykorzystywania bardziej zaawansowanych funkcji. Jednak jak na pierwszą wersję produktu robi on całkiem pozytywne wrażenie.

Warto również podkreślić to, że standardowe szablony są bardzo estetyczne. Powstające na ich bazie aplikacje HTML DB mają atrakcyjny i przyciągający uwagę wygląd. Łatwość jego zmieniania jest bardzo mocną stroną pakietu. Kilkomna dosłownie kliknięciami myszki można nadać aplikacji całkowicie inną formę graficzną. Gdy mimo wszystko żaden z dostarczonych szablonów nas nie zadawała można opracować własny, jednak to zadanie jest już dużo trudniejsze i wymaga od developera przynajmniej dobrej znajomości języka HTML.

Można również skorzystać z możliwości kopiowania szablonów między aplikacjami. Dzięki temu możemy w łatwy sposób wykorzystać stworzone przez kogoś innego rozwiązania. Bardzo ułatwia to tworzenie różnych aplikacji o ujednoczonej szacie graficznej.

6.3. Obszar zastosowania

Pakiet HTML DB należy uznać za udane przedsięwzięcie firmy Oracle, wychodzące naprzeciw bardziej konserwatywnym użytkownikom i developerom aplikacji internetowych, którzy z różnych powodów nie chcą poddać się wszechogarniającej modzie (a może konieczności ?) na aplikacje tworzone w technologiach Java.

Istnieją całkiem spore obszary, gdzie nie wydaje się ekonomicznie uzasadnione wprzęgnięcie w proces budowy aplikacji złożonych a przez to trudnych i kosztownych we wdrożeniu i utrzymaniu technologii. Chodzi tutaj głównie o proste formularze do prowadzenia i edycji danych oraz niezbyt złożone raporty z możliwością pracy *on-line* (np. sortowanie wyników według wybranej kolumny). Często otrzymany od zleceniodawcy czas na wykonanie tego typu zadań jest bardzo krótki i w takiej sytuacji pakiet HTML DB może okazać się bardzo przydatny.

Pakiet HTML DB może być godny polecenia do tworzenia interfejsów dla takich aplikacji bazodanowych jak np. listy dyskusyjne, aplikacje typu *search engine* i tym podobne. Wówczas oczywiście cała (lub większość) logiki aplikacji tworzona jest z wykorzystaniem dedykowanych do tego celu narzędzi – na przykład wyszukiwarka internetowa może korzystać bardzo intensywnie z pakietu *Oracle Text*. Być może nie wszyscy zdają sobie sprawę z tego, że bardzo popularny w świecie serwis *AskTom* (asktom.oracle.com) zbudowany został według takiej właśnie koncepcji.

Do tworzenia rzeczywiście złożonych aplikacji, których czas przewidywanego życia jest liczony co najmniej w miesiącach a najczęściej w latach, pakiet HTML DB jest już dużo bardziej kontrowersyjny i raczej autorzy artykułu nie zalecaliby jego użycia. Pierwszy powód to ten, że obecna wersja pakietu, mimo że bardzo już dopracowana, jest jednak dopiero pierwszą wersją. Gdyby w przyszłości pakiet ten był intensywnie rozwijany, z pewnością zmieni się bardzo znacznie. Drugi powód jest bardziej istotny. Otóż, mimo obecnych deklaracji firmy Oracle, dalszy rozwój pakietu nie jest definitywnie przesądzony. O ile trudno sobie obecnie wyobrazić, aby firma Oracle odstąpiła od wspierania technik opartych o języki Java oraz PL/SQL (a pewnie też trzeba tu wymienić PHP), to zdecydowanie łatwiej jest wyobrazić sobie zaprzestanie rozwijania pakietu HTML

DB. Ponieważ nie istnieje coś takiego jak plik źródłowy aplikacji HTML DB (przypomnijmy: aplikacja zapisywana jest w repozytorium, które pełni rolę metadanych, na podstawie których motor HTML DB buduje w trybie *on-line* poszczególne ekrany aplikacji), przeniesienie jej na inną platformę programistyczną jest praktycznie niewykonalne.

Powyższe stwierdzenie o być może niepewnym losie pakietu nie jest bezpodstawne. Firma Oracle swego czasu rozwijała narzędzie o nazwie *WebDB*, które potem praktycznie zniknęło z oferty firmy – oczywiście z odpowiednimi konsekwencjami dla utworzonych już w tym narzędziu aplikacji. Porównując oba narzędzia nasuwa się nieodparte skojarzenie, że *WebDB* był prekursorem HTML DB. Oba narzędzia bazują praktycznie na tym samym pomysle – nie jest wykluczone, że pewne części były tworzone przez tą samą grupę developerów i projektantów. Dziwi tylko to, że firma Oracle w żadnym miejscu nie „przyznaje” się do rodowodu HTML DB. Powodów takiego zachowania można się tylko domyślać.

Bibliografia

1. Strona domowa pakietu HTML DB:
<http://www.oracle.com/technology/products/database/htmlldb/index.html>
2. Dokumentacja systemowa: *Install and Configure Oracle HTML DB for use by a 9iR2 Oracle HTTP Server*, http://www.oracle.com/technology/products/database/htmlldb/howtos/howto_use9ir2ohs.html
3. Dokumentacja systemowa: *Install and Configure Oracle HTML DB for use with Oracle Application Server 10g*, http://www.oracle.com/technology/products/database/htmlldb/howtos/howto_useoas10g.html
4. Dokumentacja systemowa: *Oracle Database Companion CD Installation Guide 10g Release 1 (10.1.0.2.0) for Windows Part No. B13693-01*
5. Dokumentacja systemowa: *Oracle Database Companion CD Quick Installation Guide 10g Release 1 (10.1.0.2.0) for Windows Part No. B13692-01*
6. Dokumentacja systemowa: *Oracle HTML DB User's Guide Release 1.5 Part No. B10992-01*
7. Pliki html w katalogu domowym pakietu HTML DB: \marvel\doc*.html