

Oracle WebServer - architektura i administrowanie

Maciej Zakrzewicz
Instytut Informatyki Politechniki Poznańskiej
Poznań, ul. Piotrowo 3a
mzakrz@cs.put.poznan.pl

Streszczenie

W artykule poruszono zagadnienia projektowania i administrowania rozproszonymi systemami informacyjnymi opartymi na środowisku *Oracle WebServer*. Przedstawiona została architektura wewnętrzna *Oracle WebServera*, mechanizmy obsługi żądań oraz zagadnienia bezpieczeństwa bazy danych i konfiguracji parametrów systemu. Dokonano także przeglądu nowych cech kolejnych wersji produktu, głównie w odniesieniu do wersji 2.0.

Technologia WWW

Gwałtowny rozwój technologii *WWW* (*World Wide Web*) w sieci Internet, jak i wewnątrz sieci korporacyjnych spowodował pojawienie się nowej jakości w wymianie i dostępie do informacji. Łatwość rozproszonego wykorzystywania systemów informacyjnych opartych na zbiorach plików tekstowych i serwerach baz danych sprawia, że wzrasta zainteresowanie produktami umożliwiającymi budowę i zarządzanie serwerami *WWW*.

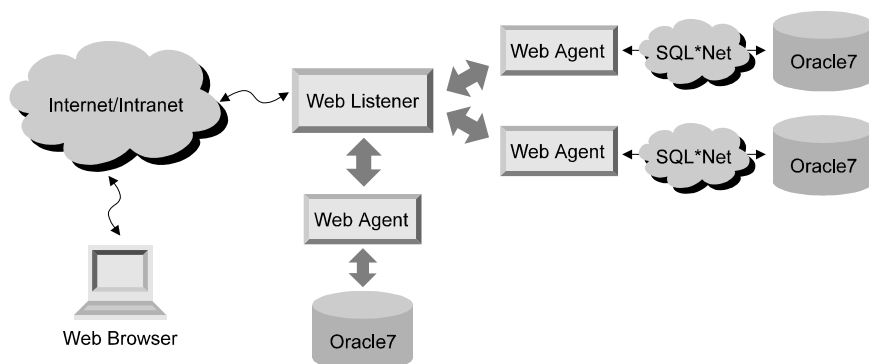
Technologia *WWW* wykorzystuje architekturę klient-serwer, w której klient (tzw. Web browser, np. Netscape, Internet Explorer) przesyła do specjalizowanego serwera komunikaty zawierające żądania. Proste żądania specyfikują dokument, który klient chce otrzymać. Dokument taki mogą stanowić informacje statyczne, jak np. kopia tekstowego raportu, lub też może on zostać wygenerowany na żądanie za pomocą zewnętrznej aplikacji pracującej po stronie serwera. Odwołania do dokumentów dostępnych poprzez *WWW* opisywane są za pomocą globalnych adresów *URL* (*Uniform Resource Locator*), zawierających: nazwę protokołu transmisji (*http*), adres IP serwera, pełną ścieżkę dostępu oraz nazwę odczytywanego pliku lub wykonywanej aplikacji zewnętrznej (np. `http://www.foo.com/pub/readme.txt`). W przypadku wywoływania aplikacji zewnętrznych, adres *URL* zawierać może także nazwy i wartości przekazywanych parametrów. Komunikacja pomiędzy klientami a serwerem odbywa się w oparciu o standard *HTTP* (*Hypertext Transfer Protocol*), który oprócz prostych żądań odczytu dokumentu specyfikuje również metody przesyłania parametrów wywoływania zewnętrznych aplikacji, autoryzacji dostępu, obsługi błędów itp.. Dokumenty są składowane i przesyłane w formacie *HTML* (*Hypertext Markup Language*), pozwalającym na kontrolowanie układu tekstu, grafiki i wideo, umieszczanie w tekście dostępnych dla użytkownika wskazań na inne części tego samego lub innych dokumentów (*hypertext link*) oraz umieszczanie elementów interakcyjnych (pola tekstowe, przyciski, listy wartości).

Istotną cechą serwerów WWW jest ich zdolność do wykonywania zewnętrznych aplikacji, dynamicznie generujących dokumenty w odpowiedzi na żądania klientów. Przykładowo, klient może wypełnić na swoim ekranie formularz *HTML*, a następnie, zatwierdzając wprowadzone dane wywołać po stronie serwera zewnętrzną aplikację do ich przetwarzania. Opracowany został standard komunikacji pomiędzy serwerem i wywoływanymi przez niego aplikacjami zewnętrznymi, którymi mogą być programy lub skrypty systemu operacyjnego (C/C++, Fortran, PERL, TCL itp.). Standard ten - *CGI* (*Common Gateway Interface*) - definiuje szereg zmiennych środowiskowych, za pomocą których serwer przekazuje parametry do aplikacji zewnętrznej. Wynik pracy aplikacji - dokument w formacie *HTML* - wysyłany jest przez nią na wyjście standardowe, a następnie przekazywany do klienta.

Oracle Web Server 1.0

Popularnym zastosowaniem aplikacji *CGI* jest dostęp za ich pomocą do serwerów relacyjnych baz danych, co umożliwia tworzenie bezobsługowych systemów informacyjnych. Działanie takich aplikacji sprowadza się do nawiązania połączenia z serwerem bazy danych, przesłania treści poleceń lub zapytań, a następnie formatowania otrzymanych wyników. Wadą takiego rozwiązania jest trudność przenoszalności i rozwoju oprogramowania. Pozytywnym w tym kontekście krokiem stało się opracowanie i wprowadzenie na rynek produktu **Oracle WebServer 1.0** (*OWS 1.0*) - kompleksowego środowiska do tworzenia i udostępniania aplikacji generujących dokumenty w oparciu o zawartość bazy danych *Oracle7* lub *Trusted Oracle7*.

Oracle WebServer 1.0 składa się z następujących elementów (rys. 1): serwera bazy danych *Oracle7*, w pełni funkcjonalnego serwera WWW (**Web Listener**), zgodnego ze standardami *HTTP/1.0* i *CGI/1.1*, specjalizowanej aplikacji *CGI* realizującej dostęp do bazy danych (**Web Agent**) oraz biblioteki wspomagającej tworzenie aplikacji przez użytkowników (**Application Developer's Toolkit**). Działanie systemu przebiega w następujący sposób. Klient wysyła do *Web Listenera* żądanie dostępu do dokumentu o określonym adresie *URL*. Na podstawie treści tego adresu *Web Listener* rozstrzyga, czy przesłany ma być dokument plikowy, czy też uruchomiony powinien zostać *Web Agent*. Uruchomiony *Web Agent* przyłącza się do serwera bazy danych i wywołuje procedurę *PL/SQL*, której wynik - dokument *HTML* - przesyłany jest za pośrednictwem *Web Listenera* do klienta, który zainicjował cały proces. Nazwa wywoływanej procedury *PL/SQL* oraz parametry jej wywołania odczytywane są z treści otrzymanego adresu *URL*.



rys.1 Architektura Oracle WebServer 1.0

Web Listener zapewnia elementarną ochronę dostępu do swoich zasobów poprzez autoryzację adresów IP oraz/lub autoryzację użytkowników. Autoryzacja adresów IP polega

na każdorazowym sprawdzaniu, czy adres IP klienta, który wysłał żądanie znajduje się na liście adresów uprawnionych do dostępu do żądanego dokumentu. Jeżeli adres IP klienta jest uprawniony, wtedy żądanie jest realizowane, jeżeli nie - wówczas do klienta wysyłany jest komunikat o odmowie dostępu. Podobnie przebiega proces autoryzacji użytkowników, przy czym użytkownik przysyła na żądanie serwera swój identyfikator i hasło dostępu. Uprzywilejowani użytkownicy *Web Listenera* przypisywani są do nazwanych grup, z kolei grupy tworzą struktury zwane zespołami (*realms*). Prawa dostępu do dokumentu specyfikowane są nie na poziomie pojedynczych użytkowników lecz na poziomie zespołów. Identyfikatory i hasła użytkowników mogą być przysyłane do serwera w sposób jawny (*basic authentication*) lub w postaci kodowanej (*digest authentication*).

Współpraca *Web Listenera* z systemem plików usprawniona jest poprzez możliwość definiowania wirtualnych systemów plików, odwzorowywanie plików w pamięci operacyjnej (*memory mapping*), buforowanie wyznaczonych dokumentów plikowych w pamięci operacyjnej oraz współdzielenie plików przez wielu klientów. Stosowanie wirtualnych systemów plików, polegające na przypisywaniu dodatkowych nazw wewnętrznym katalogom systemu plików i traktowaniu tych nazw jako katalogów wirtualnych, powoduje, poza skróceniem długości adresu *URL*, także ograniczenie dostępu klientów wyłącznie do jawnie zdefiniowanych w systemie katalogów wirtualnych. Pewnym utrudnieniem jest brak dynamicznego przeglądania zawartości katalogów, w wyniku czego *Web Listener* "nie zauważa" dodanych podczas jego pracy nowych plików dokumentów.

Dla administratora cenna jest wiedza o stopniu wykorzystywania poszczególnych dokumentów przez klientów. *Web Listener* prowadzi plikowy dziennik wszystkich otrzymanych żądań. W dzienniku tym przechowywane są: adres IP klienta, czas otrzymania żądania oraz treść żądania. Ponieważ plik dziennika zapisany jest w standardowym dla serwerów *WWW* formacie (*Common Log Format*), dlatego może on być analizowany i przetwarzany za pomocą szeregu dostępnych pakietów (*Getstats*, *WWWstat*, *Wusage* itp.). W oddzielnym pliku zapisywane są komunikaty o błędach, jakie pojawiły się w trakcie uruchamiania lub podczas pracy *Web Listenera*.

Web Listener jest programem rezydentnym, który odpowiada na żądania skierowane pod określony numer portu sieciowego komputera, na którym pracuje. Standardowo, serwery *WWW* obsługują port o numerze 80 - wówczas klient nawiązujący z nimi połączenie specyfikuje wyłącznie adres IP komputera, na którym pracuje serwer (np. `http://www.foo.com`). Jeżeli serwer *WWW* obsługuje port różny od standardowego, wtedy klient specyfikuje poza adresem IP komputera, także numer portu (np. `http://www.foo.com:8888`). Z tego względu możliwa jest równoczesna praca wielu niezależnych *Web Listenerów* na pojedynczym komputerze, pod warunkiem, że każdy z nich obsługuje oddzielny port sieciowy. Standardowa konfiguracja *Oracle Web Servera 1.0* wykorzystuje port o numerze 8888 dla zainstalowania wyróżnionego *Web Listenera*, służącego do administracji systemem.

Dla każdego *Web Listenera* zdefiniowanych może być wiele *Web Agentów* - aplikacji *CGI*, których zadaniem jest translacja treści otrzymanego adresu *URL* do sparametryzowanego wywołania rezydującej w bazie danych procedury *PL/SQL*, a następnie jej uruchomienie. W ten sposób pojedynczy *Web Listener* może uzyskiwać dostęp do wielu instancji baz danych. Uruchomiony *Web Agent* przyłącza się do serwera bazy danych za pomocą protokołu *SQL*Net V2*, uruchamia składowaną w bazie danych procedurę *PL/SQL*, wyniki wyświetlane przez procedurę przekazuje do klienta za pomocą *Web Listenera*, po czym odłącza się od serwera bazy danych. Dla każdego żądania uruchamiany jest kolejny proces *Web Agent*a. Ponieważ tworzenie nowego procesu w systemie operacyjnym jest czasochłonne, dlatego

skalowalność i przepustowość *Web Agent*a zrealizowanego w technologii CGI jest ograniczona. Wada ta usunięta została w kolejnej wersji produktu (*OWS 2.0*).

Zadaniem procedur *PL/SQL* wywoływanych przez *Web Agent*a jest generowanie treści dokumentów *HTML*. W celu uproszczenia i przyspieszenia produkcji takich procedur, do środowiska *Oracle Web Server 1.0* dołączona została biblioteka wspomagająca tworzenie procedur przez użytkowników (*Application Developer's Toolkit*). Biblioteka zbudowana jest z procedur generujących pojedyncze komendy *HTML* oraz tworzących interakcyjne elementy dokumentów. Jej zawartość może być w nieskomplikowany sposób rozszerzana przez użytkownika o dodatkowe elementy interfejsu *HTML* (standardowo - pełna zgodność z *HTML-2*). Ponieważ procedury biblioteki zostały stworzone w języku *PL/SQL*, dlatego zapewniona jest ich przenoszalność pomiędzy różnymi platformami systemowymi, dla których dostępny jest serwer *Oracle7*.

Konfiguracja i administracja

Administrowanie *Oracle Web Server 1.0* prowadzone może być za pomocą wyróżnionego *Web Listener*a (standardowo obsługującego port sieciowy o numerze 8888), do którego administrator przyłącza się za pomocą programu - klienta (*Web browser*). Tworzenie i konfiguracja parametrów *Web Listener*ów i *Web Agent*ów odbywa się interakcyjnie z wykorzystaniem dokumentów - formularzy *HTML*, dla których dostępny jest system szczegółowej pomocy kontekstowej. Parametry konfiguracyjne *Web Listener*a obejmują m.in. ustawienia sieciowe, lokalizację pliku dziennika, definicję katalogu użytkowników, ustawienia narodowe i językowe, definicje katalogów wirtualnych, specyfikację buforowanych plików, definicje użytkowników, ich grup i zespołów oraz określenie przywilejów dostępu do plików i katalogów. Z kolei ustawienia dotyczące *Web Agent*a związane są głównie ze: specyfikacją wykorzystywanej lokalnej lub zdalnej bazy danych *Oracle7*, określeniem konta użytkownika, do którego zakładane będą połączenia (*OWA Database User*), wskazaniem *Web Listener*a uprawnionego do uruchamiania tego *Web Agent*a, lokalizacją pliku dziennika oraz ustawieniami językowymi. Podczas tworzenia nowej definicji *Web Agent*a możliwe jest automatyczne utworzenie w bazie danych konta użytkownika oraz instalacja biblioteki wspomagającej tworzenie aplikacji przez użytkowników (*Application Developer's Toolkit*).

Charakterystyka dokumentów

Główną wadą dokumentów gromadzonych w plikach jest trudność związana z aktualizacją ich zawartości. Jeżeli natomiast wszystkie dane udostępniane poprzez serwer WWW pobierane będą z bazy danych, wówczas wzrośnie obciążenie serwera bazy danych. W tym kontekście wyodrębnić można cztery kategorie dokumentów, jakie mogą być udostępniane przez *Web Listener*:

- statyczne
- statyczne generowane automatycznie
- dynamiczne generowane na żądanie klienta
- dynamiczne generowane w odpowiedzi na zdarzenie

Dokumenty statyczne tworzone są przez użytkowników jako pliki w formacie *HTML*, a zmiana ich treści możliwa jest za pomocą manualnej ingerencji w zawartość pliku. Dokumenty statyczne generowane automatycznie odróżniają się tym, że ich tworzenie odbywa się w podobny sposób, jak tworzenie raportów drukowanych - np. za pomocą narzędzi *Developer/2000*. Technika ta pozwala na wykorzystanie istniejących aplikacji/raportów dla wzbogacenia systemu informacyjnego opartego na *Oracle Web Server*. Dokumenty dynamiczne generowane na żądanie klienta realizowane są przez procedury *PL/SQL*

zgrupowane w bazie danych. Wywoływaniem i przekazywaniem parametrów do tych procedur zajmuje się *Web Agent*. Z kolei dokumenty dynamiczne generowane w odpowiedzi na zdarzenie (np. w regularnych odstępach czasu) są faktycznie statycznymi replikami dokumentów dynamicznych. Dokumenty takie znajdują zastosowanie w systemach o dużym obciążeniu serwera bazy danych i niskiej zmienności publikowanych informacji (żądania klientów nie są realizowane w czasie rzeczywistym przez *Web Agent*). Tworzeniem dokumentów dynamicznych generowanych w odpowiedzi na zdarzenie mogą zajmować się skrypty systemu operacyjnego. Podział publikowanych informacji pomiędzy wymienione kategorie dokumentów jest jednym z elementów projektowania efektywnego systemu informacyjnego opartego na *Oracle WebServer*.

Bezpieczeństwo

W przypadku serwerów WWW, które udostępniają informacje poufne lub pozwalają klientom na wprowadzanie prywatnych informacji do bazy danych kluczowym zagadnieniem jest bezpieczeństwo danych i ich transmisji. Szczególnie narażone na próby nieuprawnionego dostępu jest konto użytkownika bazy danych, do którego przyłącza się *Web Agent*. Zagrożone są dane, kody źródłowe procedur *PL/SQL* generujących dokumenty oraz sam serwer bazy danych. Użytkownik *Web Agent* powinien posiadać minimalne uprawnienia oraz tylko niezbędne do prawidłowego działania obiekty. Zalecane jest stosowanie podwójnych użytkowników: pierwszy (strzeżony) jest właścicielem relacji lub perspektyw z danymi, procedur generujących dokumenty oraz biblioteki wspomagającej tworzenie procedur użytkownika, drugi (ogólnodostępny użytkownik *Web Agent*) - posiada wyłącznie przywilej nawiązywania połączenia oraz wykonywania procedur generujących dokumenty, będących wtedy własnością pierwszego użytkownika. W takiej konfiguracji, znajomość hasła dostępu do użytkownika *Web Agent* przez osobę nieuprawnioną, nie narusza praktycznie bezpieczeństwa bazy danych. Należy jednocześnie pamiętać, iż informacje niezbędne do przyłączenia się do bazy danych przechowywane są w plikach konfiguracyjnych *Web Agent* i dlatego powinny one być zabezpieczone przy pomocy mechanizmów systemu operacyjnego.

Oracle Web Server 1.0 umożliwia definiowanie użytkowników, grup użytkowników i zespołów (zespół jest zbiorem grup) w celu autoryzacji dostępu do wybranych dokumentów. Ponieważ liczba takich użytkowników jest ograniczona (oraz niewielka), opis każdego użytkownika bardzo skrótowy, a zmiana hasła dostępu dla pojedynczego użytkownika musi być dokonywana przez administratora, dlatego ta metoda autoryzacji może pełnić jedynie funkcję prostego ograniczania dostępu. W systemach obsługujących setki i tysiące użytkowników proponuje się metody autoryzacji użytkowników na poziomie procedur *PL/SQL*. Informacje o wszystkich użytkownikach *Oracle Web Servera* gromadzone są wówczas w bazie danych, a stosowna procedura *PL/SQL* przeprowadza weryfikację użytkownika i kontroluje jego dostęp do określonych dokumentów. Dzięki temu użytkownicy mogą swobodnie modyfikować swoje hasła dostępu, możliwe jest nawet automatyczne tworzenie ich kont. Oczywiście autoryzacja na poziomie procedur *PL/SQL* dotyczyć może tylko dostępu do innych procedur, a nie do dokumentów statycznych, co nie wydaje się jednak być istotnym ograniczeniem (dokument statyczny może być przechowywany w bazie danych). Luką w bezpieczeństwie tej metody autoryzacji jest niekodowana transmisja danych pomiędzy serwerem i klientem. Dopiero *Oracle Web Server 2.0* gwarantuje bezpieczeństwo transmisji poprzez implementację standardu *SSL 2.0*.

Dziennik pracy systemu

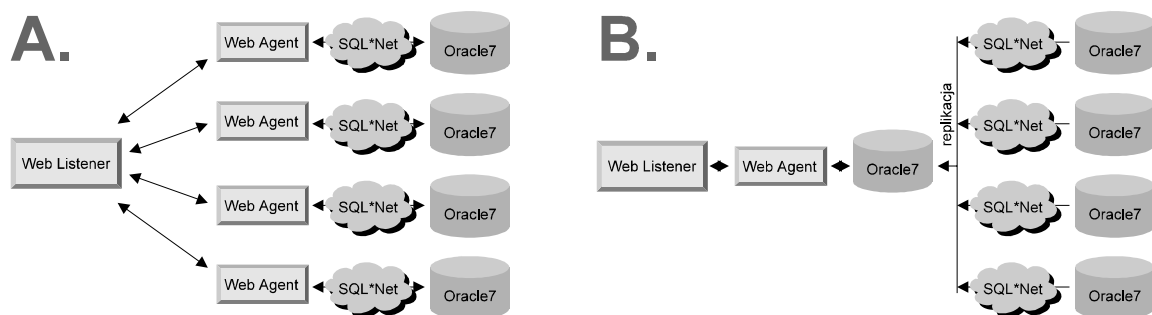
Zbiór informacji o wykorzystaniu serwera oraz jego poszczególnych dokumentów gromadzony jest w pliku dziennika (*info file*). Ponadto w pliku błędów (*error file*),

zapisywane są komunikaty o błędach, jakie wystąpiły w czasie pracy *Web Listenera*. Wiedza zgromadzona w tych plikach jest wykorzystywana do strojenia i rozbudowy serwera. Pakiet *Oracle Web Server 1.0* nie zawiera wprawdzie narzędzi ułatwiających analizę pliku dziennika, ale ponieważ plik ten zapisywany jest w standardowym dla serwerów WWW formacie (*Common Log Format*), dlatego jego przetwarzanie może być prowadzone za pomocą łatwo dostępnych pakietów, takich jak *Getstats*, *WWWstat* i *Wusage*.

Jeżeli informacje o zewnętrznych dostęпах do serwera są szeroko wykorzystywane, wówczas proponuje się stworzenie dodatkowego, proceduralnego dziennika aktywności systemu. Dziennik proceduralny składa się ze zbioru relacji przechowujących informacje o połączeniach oraz ze specjalizowanych procedur *PL/SQL*, które wprowadzają i analizują dane z tych relacji. Ponadto, z wywołaniem każdej procedury generującej dokumenty należy wiązać wywołanie procedury aktualizacji dziennika. Zaletą takiego rozwiązania jest mniejsza objętość dziennika (powtarzające się zapisy są efektywnie przechowywane), selekcja gromadzonych informacji (np. tylko połączenia z określoną domeną sieciową), szybkość wyszukiwania informacji (serwer bazy danych) oraz możliwość prostego wykorzystania tych danych w innych aplikacjach (np. *Developer/2000*).

Architektura zewnętrzna

Oracle Web Server 1.0 może współpracować zarówno z lokalnymi, jak i zdalnymi serwerami bazy danych *Oracle7*. Korzystne w kontekście izolacji systemów zdalnych baz danych oraz nie wpływania na ich obciążenie może być tworzenie lokalnych replik wszystkich danych udostępnianych przez *Oracle WebServer* (rys. 2). Odwołania *Web Agent* przetwarzane są wtedy lokalnie, np. na dedykowanym komputerze-serwerze. Problemy takich rozwiązań skupiają się wokół konieczności utrzymywania replik, trudności w dostarczaniu aktualnych danych oraz skomplikowanego mechanizmu modyfikacji zawartości bazy danych przez klientów WWW (konieczność dostępu do relacji źródłowych). Dla systemów informacyjnych ukierunkowanych na świadczenie usług w pełni poprzez WWW zalecana jest architektura rozproszona, w której odwołania *Web Agentów* przetwarzane są przez wiele zdalnych serwerów baz danych *Oracle7*.



rys.2 Zewnętrzne architektury Oracle WebServera: rozproszona (A) i lokalna (B)

Oracle Web Server 2.0 i wersje kolejne

Wiele wad systemu *Oracle Web Server 1.0* zostało wyeliminowanych w nowej wersji produktu - *Oracle Web Server 2.0*. Różnice w architekturze systemu skupiają się głównie na zastąpieniu interfejsu CGI stosowanego dotychczas pomiędzy *Web Listenerem* a *Web Agentem* przez moduł **Web Request Broker (WRB)**, sterujący przekazywaniem żądań do innych modułów aplikacyjnych oraz na wprowadzeniu interfejsu **WRB API (Web Request Broker Application Programmer's Interface)**, umożliwiającego tworzenie przez użytkownika

własnych modułów aplikacyjnych. Pozostałe cechy nowej wersji *Oracle Web Servera* to również: trwałe połączenie z bazą danych, dynamiczne równoważenie obciążenia, dynamiczne przeglądanie zawartości wirtualnych katalogów, bezpieczna transmisja od/do klienta z wykorzystaniem *SSL 2.0 (Secure Socket Layer)*, interpreter języka *Java* po stronie serwera oraz przetwarzanie dokumentów przez serwer - tzw. *Server Side Includes (SSI)*.

Całkowicie nowy moduł - *Web Request Broker* zajmuje się przetwarzaniem żądań wysyłanych przez klientów do *Web Listenera*. *Web Request Broker* składa się z: elementu sterującego - tzw. *WRB Dispatcher (WRBD)* oraz z wielu elementów obsługujących - *WRB Execution Engine (WRBX)*. Element sterujący (*WRBD*) otrzymuje od *Web Listenera* treść żądania wysłanego przez klienta i na podstawie własnych plików konfiguracyjnych określa typ żądanego obiektu (np. aplikacja *Java*, procedura *PL/SQL*), tym samym identyfikuje element obsługujący (*WRBX*), który zajmie się dalszym przetwarzaniem (np. interpreter języka *Java*, *PL/SQL Agent*). Element obsługujący (*WRBX*) wywołuje właściwą aplikację zewnętrzną (np. *PL/SQL Agenta*) i przekazuje jej parametry odczytane z żądania wysłanego przez klienta. Elementy obsługujące (*WRBX*) w *Oracle Web Server 2.0* potrafią współpracować z *PL/SQL Agentem* (odpowiednik *Web Agent* w *OWS 1.0*), interpreterem języka *Java* oraz innymi modułami, które wykorzystują interfejs *WRB API*.

Zastosowany w *Oracle Web Server 2.0* moduł *PL/SQL Agent* realizuje takie same funkcje jak *Web Agent* wykorzystany w poprzedniej wersji systemu. Nowy *PL/SQL Agent* pracuje szybciej niż jego poprzednia wersja, czego głównym powodem jest utrzymywanie trwałego połączenia z serwerem bazy danych oraz zastąpienie interfejsu *CGI* poprzez specjalizowany moduł *Web Request Broker*.

Kolejne wersje produktu *Oracle Web Server* wyposażone zostaną w możliwości współpracy z innymi dostępnymi na rynku serwerami *WWW (OWS 2.1 z Netscape HTTP Server, OWS 3.0 z Microsoft Internet Information Server)*, zdolność przechowywania w bazie danych wszystkich obiektów związanych z serwerem (dokumenty statyczne, grafika, aplikacje *Java* itp. w *OWS 3.0*), pełną implementację serwera *Proxy* (buforowanie pomiędzy klientem a serwerem) itd.

Perspektywy rozwoju

Obserwacja rynku pozwala dostrzec w technologii *WWW* przyszłość dla klasycznych systemów klient-serwer. Wśród przesłanek ekonomicznych istotne wydaje się być zmniejszenie kosztów instalacji i utrzymywania oprogramowania, gdyż, w przeciwieństwie do systemów klient-serwer, w technologii *WWW* operacje te nie dotyczą setek komputerów osobistych, lecz wyłącznie kilku lub kilkunastu serwerów. Ponadto, łatwość wykorzystywania aplikacji *WWW* przez użytkowników, wynikająca z jednolitości interfejsu, pociąga za sobą redukcję kosztów szkolenia. Uniwersalny dostęp do aplikacji (*Web browser*) oraz zastosowanie otwartych standardów, nie narzuconych przez pojedynczego producenta, mogą sprawić, że *WWW* stanie się jedną z popularniejszych platform aplikacyjnych.

Materiały źródłowe:

Oracle - "*Oracle WebServer 1.0 - User's Manual*", dokumentacja Oracle WebServer 1.0

Oracle - "*Oracle WebServer 2.0*", Technical Note, 03.1996

Oracle - "*Oracle Intranet Strategy*" - An Oracle White Paper, 06.1996

D.R.T. Robinson - "*The WWW Common Gateway Interface Version 1.1*", Internet Draft, 15.02.1996

T. Berners-Lee, R. Fielding, H. Frystyk - "*Hypertext Transfer Protocol -- HTTP/1.0*", Internet Draft, 19.02.1996