

XV Seminarium PLOUG
Warszawa
Maj 2007

OAS Web Cache: przegląd zastosowań

Maciej Zakrzewicz
PLOUG

mzakrzewicz@cs.put.poznan.pl

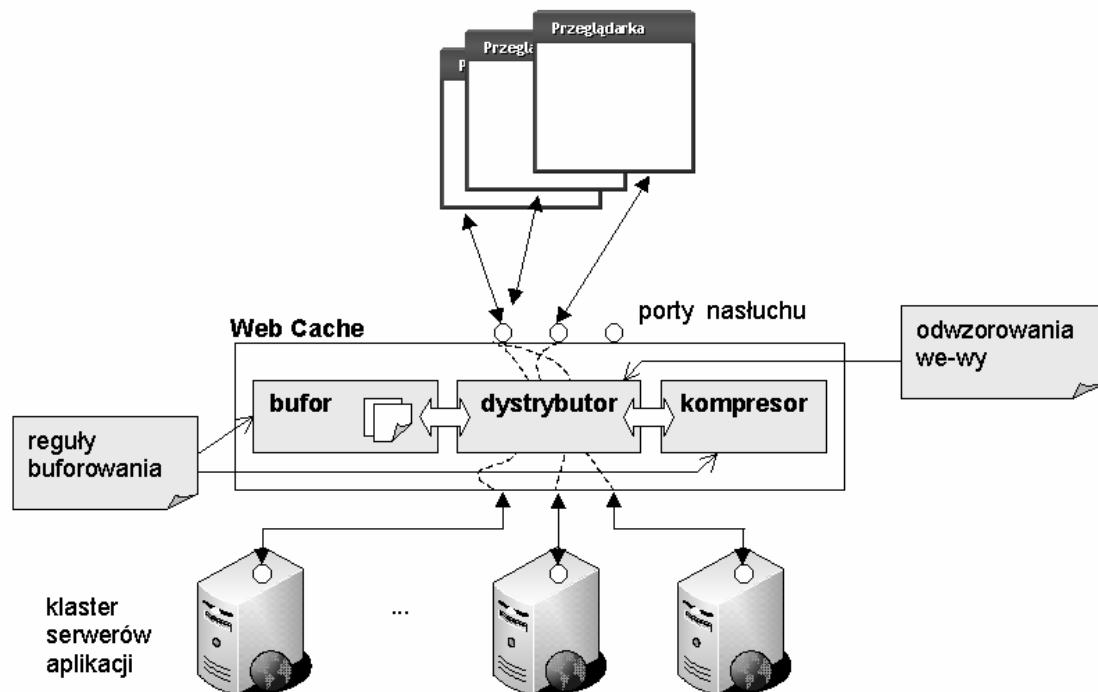
Abstrakt. Web Cache jest jednym z produktów rodziny Oracle Application Server 10g, oferującym mechanizmy buforowania odpowiedzi HTTP, równoważenia obciążenia i kompresji danych. Najczęściej jest wykorzystywany w celu optymalizacji wydajności środowiska serwera aplikacji. Celem referatu jest omówienie wewnętrznej architektury OAS Web Cache, przedstawienie scenariuszy wybranych zastosowań oraz omówienie zasad administracji.

Informacja o autorze. Pracownik Instytutu Informatyki Politechniki Poznańskiej oraz prezes Zarządu Stowarzyszenia Polskiej Grupy Użytkowników Systemu Oracle. Zainteresowania naukowe obejmują eksplorację danych (data mining), systemy baz danych/hurtowni danych oraz technologie internetowe. Dla krajowych i zagranicznych uniwersytetów oraz przedsiębiorstw (m.in. Niemcy, Wielka Brytania, USA, Słowacja, Słowenia) prowadzi wykłady i szkolenia z zakresu projektowania i implementacji systemów informatycznych. Kieruje i doradza w projektach informatycznych realizowanych w architekturach internetowych. Autor ponad 100 publikacji z zakresu odkrywania zbiorów częstych, przetwarzania zapytań eksploracyjnych, adaptatywnych serwerów WWW, technik indeksowania baz danych, strojenia systemów baz danych, standardów Java dla baz danych i dla aplikacji wielowarstwowych. Współtwórca serii konferencji i seminariów PLOUG, służących transferowi wiedzy i technologii wśród krajowych przedsiębiorstw informatycznych.

Wprowadzenie

Serwer Web Cache jest jednym z czterech produktów (obok Middle Tier, Identity Management i Metadata Repository) wchodzących w skład rodziny Oracle Application Server 10g. Jego podstawowa rola polega na odciążeniu serwera aplikacji (Middle Tier) poprzez buforowanie dokumentów wynikowych dla najczęściej przekazywanych żądań. Dzięki buforowaniu, żądania przekazywane przez przeglądarki internetowe użytkowników końcowych mogą być automatycznie obsłużone przez Web Cache i nie docierają do serwera aplikacji. Web Cache może również pełnić rolę dystrybutora żądań w klastrach serwerów aplikacji. Wówczas żądania użytkowników trafiają do Web Cache, a następnie są przekazywane do różnych węzłów klastra (algorytm Round Robin), zapewniając jego transparentność. Dokumenty wynikowe zwracane przez Web Cache mogą być opcjonalnie kompresowane, w celu redukcji zapotrzebowania na pasmo sieciowe.

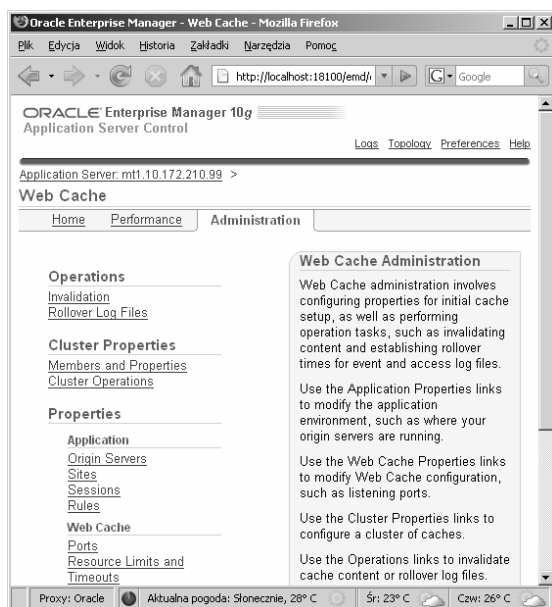
Podstawowa architektura serwera Web Cache została przedstawiona na rys. 1. Web Cache składa się z trzech głównych modułów: bufora dokumentów, dystrybutora żądań i kompresora dokumentów. Przeglądarki internetowe wysyłają żądania HTTP do portów nasłuchu serwera Web Cache. Po otrzymaniu każdego żądania, Web Cache sprawdza, czy posiada już w pamięci buforowej kopię dokumentu wynikowego dla identyfikatora URL żądania. Jeżeli tak, to natychmiast udziela odpowiedzi przeglądarce korzystając z kopii dokumentu z bufora. W przeciwnym przypadku, Web Cache generuje wtórne żądanie dla serwera aplikacji, pobiera dokument, którego domaga się przeglądarka, warunkowo umieszcza otrzymany dokument w pamięci buforowej, a następnie odsyła do przeglądarki. O tym, czy pobrany dokument będzie buforowany, decydują tzw. reguły buforowania, definiowane przez administratora. Web Cache zastosowany w środowisku klastra serwerów aplikacji może automatycznie rozdzielać otrzymywane żądania pomiędzy serwery aplikacji. Zasady równoważenia obciążenia są definiowane przez administratora w formie tzw. odwzorowań wejścia-wyjścia.



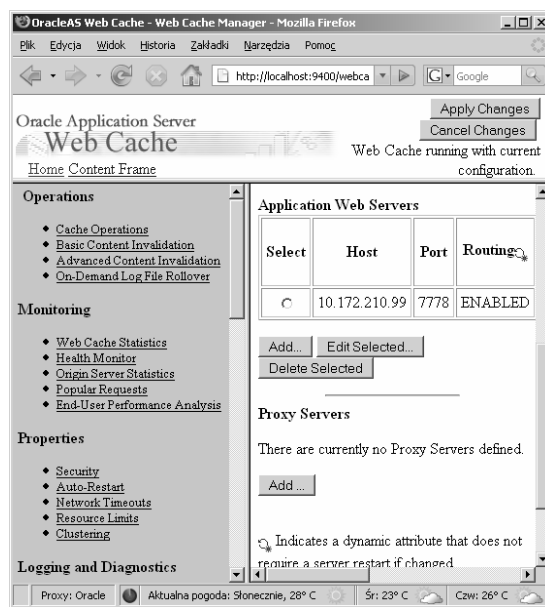
Rys. 1. Architektura serwera Web Cache

Narzędzia administracyjne

Do zarządzania serwerem Web Cache mogą być wykorzystane cztery narzędzia: aplikacja WWW Application Server Control, aplikacja WWW Web Cache Manager, skrypt systemowy „opmncctl” oraz skrypt systemowy „webcachectl”. Każde z nich umożliwia uruchomienie, zatrzymanie oraz obserwację stanu pracy serwera Web Cache. Tylko narzędzia WWW pozwalają na manipulowanie parametrami konfiguracyjnymi. Narzędzie Application Server Control zilustrowano na rys. 2, a Web Cache Manager – na rys. 3.



Rys. 2. Narzędzie Application Server Control



Rys. 3. Narzędzie Web Cache Manager

Reguły buforowania

O umieszczaniu dokumentów wynikowych w buforze Web Cache decydują definiowane przez administratora **reguły buforowania** (ang. caching rules, cacheability rules). Każda reguła buforowania jest oparta na adresie URL dokumentu i określa, czy dokument będzie buforowany, jeśli tak to przez jak długi okres czasu i czy będzie kompresowany przed wysłaniem do przeglądarki użytkownika końcowego. Poniżej przedstawiono przykład prostej reguły buforowania:

```
url=~\.(\.gif|jpe?g)$ ^ method=get => cache=yes ^ compression=off ^ expiration=5min
```

Zgodnie z tą regułą, dokumenty wynikowe pobierane metodą HTTP GET, których adres URL spełnia wyrażenie regularne „\.(gif|jpe?g)\$” (kończy się słowem „gif”, „jpeg” lub „jpg”), są umieszczane w buforze na okres 5 minut i są następnie udostępniane w postaci niekompresowanej. Reguły buforowania są definiowane przez administratora za pomocą narzędzi Application Server Control lub Web Cache Manager. Na rys. 4 przedstawiono definicję wspomnianej reguły buforującej dokonaną za pomocą Application Server Control.

General		Advanced Caching Instructions	
Name	Buforuj obrazki	Site	All Sites
Description	Reguła powoduje buforowanie plików graficznych .gif, .jpg		<input checked="" type="checkbox"/> Enabled
Selector			
To match this rule, a request must match all the parts of the Selector.			
Match URL By	Regular Expression	*\.(\.gif jpe?g)\$	
▶ Show HTTP Methods and Parameters			
Instructions			
Caching			
<input checked="" type="radio"/> Cache	Expiration of Cached Response	Expires: Max Time in Cache, 5 Minutes, Refresh: Immediately	
<input type="radio"/> Do not cache	Related Link	Expiration Policies	
Compression			
Web Cache can compress responses whether or not they are cached. Do not compress objects which are already compressed, such as GIF and JPEG files. Netscape 4.x browsers are unable to uncompress files which are included in HTML content, such as JavaScript files.			
<input checked="" type="radio"/> Do not compress			
<input type="radio"/> Compress for all browsers			
<input type="radio"/> Compress for all browsers except Netscape 4.x			

Rys. 4. Definicja reguły buforowania

Na szczególną uwagę zasługują reguły buforowania dokumentów dynamicznych, generowanych przez aplikację internetową (np. serwlet Java, strona JSP, aplikacja Oracle Reports, strona PHP) w odpowiedzi na żądanie przeglądarki. W przypadku dokumentów dynamicznych, adres URL niekoniecznie jednoznacznie identyfikuje ich treść. Treść dokumentów dynamicznych może bowiem być konstruowana z uwzględnieniem dodatkowych parametrów żądania, zmiennych Cookies, itp. Przykładowo, klienci banku internetowego wysyłają żądania pobrania dokumentów przedstawiających salda kont korzystając z jednakowego adresu URL, a mimo to każdy z nich uzyskuje dokument o innej treści, gdyż bankowa aplikacja internetowa oprócz adresu URL bierze też pod uwagę zmienną Cookie identyfikującą sesję użytkownika, który wystosował żądanie. Gdyby serwer Web Cache nie posiadał wiedzy o niejednoznaczności adresu URL, mogłoby się zdarzyć, że dokument opisujący saldo konta jednego klienta byłby udostępniany pozostałym klientom jako buforowana kopia ich dokumentów sald. W celu uniknięcia zagrożeń wynikających z niejednoznaczności adresów URL, reguły buforowania mogą dodatkowo odwoływać się do wartości zdefiniowanych zmiennych Cookies, pól nagłówkowych HTTP (np. pola Accept-language wskazującego preferowany język narodowy użytkownika przeglądarki), identyfikatora sesji HTTP czy też zawartości ciała żądania HTTP POST.

Dokumenty wynikowe są przechowywane w buforze tylko przez określony czas. Po upływie tego czasu, dokument może bądź zostać usunięty z bufora, bądź też automatycznie odświeżony w wyniku żądania samodzielnie ponowionego przez serwer Web Cache. Automatyczne odświeżanie jest realizowane przez Web Cache w dość ostrożny sposób tak, aby nie powodować zagrożeń dla wydajności serwerów aplikacji. Między innymi z tego powodu administrator może określić przedział czasowy - a nie punkt czasowy, w którym powinno dojść do odświeżenia dokumentu.

Unieważnianie zawartości bufora

Dokumenty wynikowe umieszczone w buforze mogą być z niego usunięte przed upływem czasu ważności deklarowanego przez regułę buforowania. Czynność taką nazywa się **unieważnianiem zawartości bufora** (ang. cache invalidation). Unieważnianie może być przeprowadzone manualnie przez administratora lub automatycznie przez aplikację zewnętrzną. W celu unieważnienia wybranych dokumentów zgromadzonych w buforze, należy skorzystać ze specjalizowanego protokołu sieciowego ESI, opartego na HTTP (metoda HTTP POST). Przykładowy komunikat unieważniający ESI został przedstawiony poniżej:

```
1  POST /x-oracle-cache-invalidate HTTP/1.0
2  Authorization: Basic aW52YWxpZGF0b3I6aW52YWxpZGF0b3I=
3  Content-Length: 217
4
5  <?xml version="1.0" ?>
6  <!DOCTYPE INVALIDATION SYSTEM "invalidation.dtd">
7  <INVALIDATION VERSION="WCS-1.0">
8  <OBJECT>
9  <BASICSELECTOR URI="/cache.htm" />
10 <ACTION />
11 </OBJECT>
12 </INVALIDATION>
```

W wierszu 1 znajduje się nagłówek komunikatu HTTP POST. Wiersz 2 zawiera nazwę i hasło użytkownika uprawnionego do unieważniania zawartości bufora, zakodowane w formacie Base64 (predefiniowany użytkownik INVALIDATOR). Wiersz 3 określa rozmiar całego komunikatu. Wiersz 4 jest separatorem. W wierszu 5 znajduje się znacznik rozpoczynający dokument XML, po którym w wierszu 6 znajduje się deklaracja typu dokumentu. Wiersze 7-12 opisują żądanie unieważnienia znajdującej się w buforze kopii dokumentu „/cache.htm”.

Komunikaty unieważniające mogą być generowane przez administratora posługującego się narzędziami Web Cache Manager (rys. 5) lub Application Server Control, przez administratora posługującego się programem Telnet, przez aplikację zewnętrzną, wyzwalacz bazodanowy (ang. trigger), itp. Mogą także być zagnieżdżane wewnątrz przesyłanych dokumentów, np. za pomocą znaczników JESI. Interesującym rozwiązaniem jest powiązanie unieważniania zawartości bufora ze zmianami zachodzącymi w bazie danych wykorzystywanej przez aplikację generującą dokumenty. Wyzwalacz reagujący na modyfikacje rekordów tabeli mógłby wysyłać do serwera Web Cache komunikaty unieważniające wszystkie raporty biznesowe, których treść została wygenerowana w oparciu o tę tabelę. Dzięki temu, uniknęlibyśmy sytuacji, w której użytkownicy otrzymują od serwera Web Cache stare kopie dokumentów wynikowych oparte na nieaktualnych już danych.

Basic Content Invalidation

Use this page to remove objects from the cache based on a URL.

For Cache:

Search Criteria

Remove all cached objects

Enter exact URL for removal:

Action

Preview list of objects that match invalidation criteria

Object Range: From To

Remove immediately

Remove objects no later than: after submission

Rys. 5. Manualne unieważnianie zawartości bufora

Dystrybucja żądań

Serwer Web Cache może zostać użyty w środowisku klastra serwerów aplikacji w charakterze **dystrybutora żądań** (ang. load balancer). Wówczas jego rola polega na odbieraniu żądań przybywających od przeglądarek internetowych, a następnie przekazywaniu ich do wykonania kolejno wybranym (algorytm Round Robin) serwerom aplikacji. Mechanizm dystrybucji żądań może funkcjonować w powiązaniu z buforowaniem dokumentów wynikowych – wtedy dystrybucja żądań będzie dotyczyć wyłącznie tych żądań, które nie mogą być obsłużone na podstawie aktualnej zawartości bufora.

Dystrybucja żądań pomiędzy serwerami aplikacji może być równomierna lub nierównomierna. Administrator określa wagę i pojemność każdego serwera aplikacji (ang. capacity), a wtedy Web Cache rozdziela żądania w sposób proporcjonalny. Przykładowo, w przypadku klastra trzech serwerów aplikacji o pojemnościach odpowiednio, 50, 50 i 100, pierwsze dwa serwery aplikacji będą otrzymywać od Web Cache po 25% żądań, a trzeci – 50%. Jednocześnie liczba równocześnie przetwarzanych żądań nie będzie mogła przekroczyć zadanych wartości (50,50,100).

W celu skorzystania z mechanizmu dystrybucji żądań, administrator definiuje tzw. **odwzorowania wejścia-wyjścia** (ang. site to server mappings), które określają adresy i rozkłady dystrybucji żądań przybywających do wskazanych portów nasłuchu serwera Web Cache. Poniżej przedstawiono przykład prostej definicji odwzorowań wejścia-wyjścia:

```
hostname=wcache1 ^ port=80 => hostname=mt1 ^ port=7777 ^ capacity=50
hostname=wcache1 ^ port=80 => hostname=mt2 ^ port=7777 ^ capacity=50
hostname=wcache1 ^ port=8080 => hostname=mt3 ^ port=8080 ^ capacity=10
```

Zgodnie z tymi odwzorowaniami, żądania przybywające do portu 80 i adresu „wcache1” serwera Web Cache są równo rozdzielane pomiędzy dwa serwery aplikacji: „mt1” port 7777 i „mt2” port

7777. Z kolei żądania przybywające do portu 8080 i adresu „wcache1” są przekazywane do serwera aplikacji „mt3” port 8080. Odwzorowania wejścia-wyjścia mogą być definiowane za pomocą narzędzi Application Server Control (rys. 6) lub Web Cache Manager.

Zadaniem dystrybutora żądań jest także monitorowanie stanu serwerów aplikacji, do których przekazywane są żądania. Jeżeli serwer aplikacji przestaje udzielać odpowiedzi, Web Cache uznaje go za uszkodzony i przestaje kierować do niego dalsze żądania. Jednocześnie, w regularnych okresach czasu wysyła próbne żądania w celu wykrycia ponownej dostępności serwera aplikacji. Administrator serwera Web Cache może określić zarówno liczbę kolejno nieudanych żądań, po przekroczeniu której serwer aplikacji uznawany jest za uszkodzony (ang. failover threshold), jak i częstotliwość późniejszego próbkowania jego dostępności (ang. ping interval).

Sites

A host and port comprise a site. Using this information, you can create a site definition. A site definition determines how Web Cache forwards a request to a given origin server. Page Refreshed May 22, 2007 4:56:48 PM

Named Sites Definitions

Create a named site to define site-specific caching rules, error pages, sessions, or aliases. You can also view performance metrics for named sites.

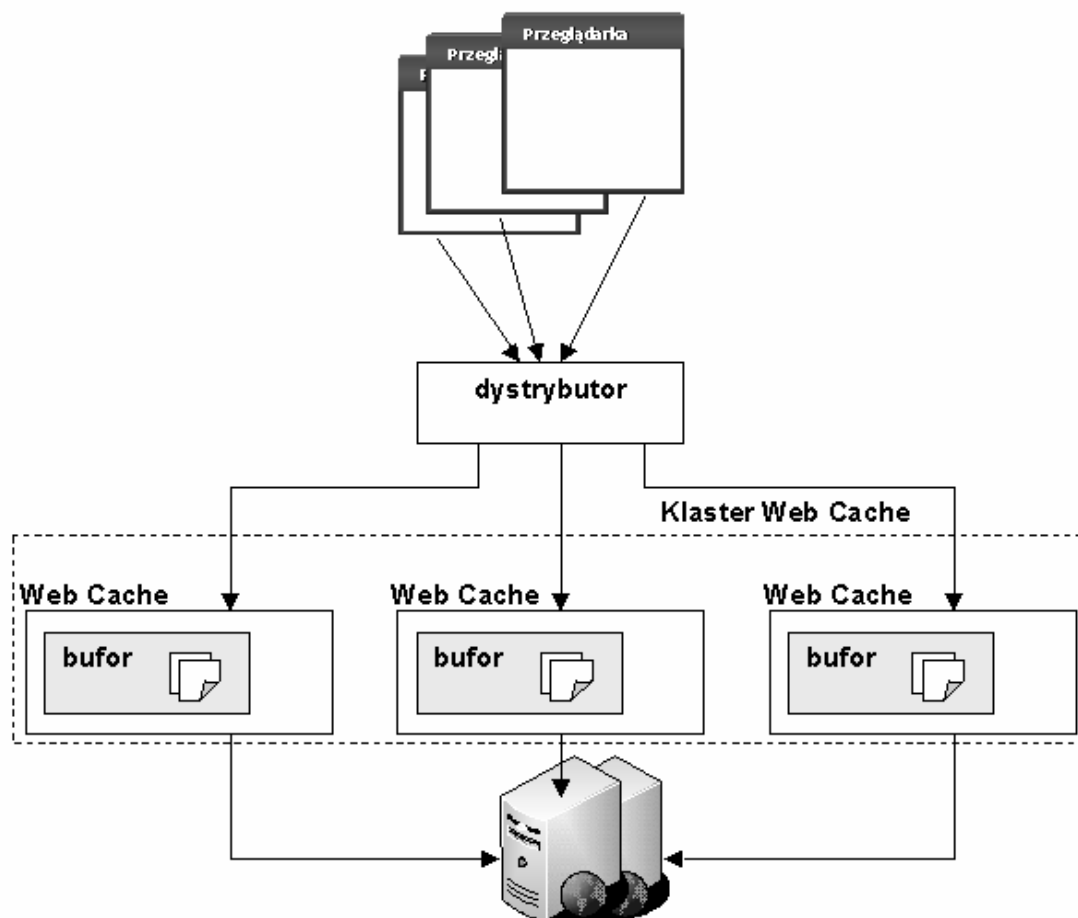
View Columns General

Select	Site	Default Site	Aliases	Origin Servers
<input type="radio"/>	wcache1:80	✓		10.10.10.10:7777 10.10.10.20:7777
<input checked="" type="radio"/>	wcache1:8080			10.10.10.30:8080

Rys. 6. Definicja odwzorowań wejścia-wyjścia

Klastry serwerów Web Cache

Interesującą cechą serwerów Web Cache jest możliwość ich rozproszonej instalacji w układzie klastra składającego się z dwóch lub większej liczby węzłów. Każdy węzeł obsługuje tylko część żądań użytkowników, lecz jednocześnie współpracuje z pozostałymi węzłami. Przykładową architekturę klastra serwerów Web Cache przedstawiono na rys. 7.



Rys. 7. Klaster serwerów Web Cache

Żądania przybywające od przeglądarek są kierowane przez dystrybutor do losowo wybranych serwerów Web Cache. Każdy serwer Web Cache obsługuje żądanie poprzez przeszukanie własnego bufora, a w przypadku niepowodzenia, poprzez przekierowanie żądania do innego serwera Web Cache – tzw. serwera odpowiedzialnego za dany adres URL. Odpowiedzialność ta jest przydzielana na podstawie wartości funkcji haszującej wyliczanej z użyciem adresu URL. Jeżeli serwer odpowiedzialny również nie posiada kopii dokumentu wynikowego w swojej pamięci buforowej, to żądanie jest kierowane do serwera aplikacji.

Konfiguracja klastrów serwerów Web Cache może być przeprowadzona przy użyciu narzędzia Application Server Control lub Web Cache Manager. Klastry serwerów Web Cache mogą służyć do zwiększenia wydajności systemu lub do zwiększenia jego odporności na awarie.

Podsumowanie

Produkt Web Cache znajduje liczne zastosowania w środowiskach zorientowanych na skalowalność i niezawodność. Dzięki wbudowanym mechanizmom buforowania może służyć do odciążenia serwera aplikacji poprzez eliminację generowania dokumentów „popularnych”. Dzięki wbudowanemu modułowi dystrybutora, może stanowić interesującą alternatywę dla oprogramowania klasy „load balancer”. Niestety, korzyści wynikające ze stosowania serwera Web Cache są zależne od specyfiki oprogramowania aplikacyjnego oraz dynamiki dokumentów wynikowych generowanych w odpowiedzi na żądania użytkowników.

Literatura

1. "Oracle Application Server 10g Administrator's Guide", Oracle
2. „Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, Fielding, et al.
3. „ESI Language Specification 1.0", Tzimelzon, Weihl, Jacobs, Akamai Technologies, Oracle Corporation
4. „ESI Invalidation Protocol 1.0", Tzimelzon, Weihl, Jacobs, Akamai Technologies, Oracle Corporation
5. „JESI Tag Library 1.0 Specification: Tags for Edge-Side Includes in JSP", Basu et al.