



Poznańskie Centrum Superkomputerowo - Sieciowe

PROJEKT LDAP

**Raport z wdrożenia :
przeniesienia informacji o użytkownikach aplikacji portalowej
z relacyjnej bazy danych (Oracle) do bazy LDAP-owej**

**Jerzy Chochulski
Mirośław Czyrnek
Jarosław Głapski**

Poznań 2003

Spis treści

| | | |
|-----|---------------------------------|---|
| 1 | Wstęp | 2 |
| 2 | Procedura przygotowawcza | 2 |
| 3 | Etap badawczy | 2 |
| 3.1 | Iteracja elementów wyniku | 2 |
| 3.2 | Złożone zapytania | 3 |
| 3.3 | Sortowanie wyniku..... | 4 |
| 3.4 | Indeksy | 4 |
| 4 | Podsumowanie..... | 5 |
| | Bibliografia | 6 |

1 Wstęp

W Poznańskim Centrum Superkomputerowo-Sieciowym rozwijany jest uniwersalny pakiet portalowy *Joshua*, implementowany w języku *Java*.

Dotychczas pakiet *Joshua* wykorzystywał do przechowywania informacji o użytkownikach relacyjną bazę danych. Ze względu na perspektywę wdrożenia pakietu w kilku współpracujących ze sobą portalach oraz wynikającą z tego potrzebę współdzielenia danych przez grupę portali, podjęto próbę przeniesienia danych o użytkownikach do struktury LDAP-owej [1].

Projekt przeniesienia wymagał opracowania odpowiedniej struktury LDAP-owej oraz rozwinięcia interfejsu dostępowego pozwalającego na korzystanie z niej.

Opracowano również narzędzie do przenoszenia danych pomiędzy strukturą relacyjną a katalogową [2]. Narzędzie wyposażono w funkcję anonimizacji przenoszonych danych w celu zapewnienia ich ochrony w myśl ustawy [3].

Końcowym etapem przedsięwzięcia było przetestowanie i wdrożenie opracowanych rozwiązań. W tym celu uruchomiono środowisko testowe i opracowano zestaw testów wydajnościowych. Niniejszy raport podsumowuje doświadczenia i wnioski z przedsięwzięcia.

2 Procedura przygotowawcza

W celu zastosowania LDAP-owej wersji narzędzia do zarządzania użytkownikami podjęto następujące kroki:

- opracowano strukturę katalogową pozwalającą na odwzorowanie danych o użytkownikach portali;
- wdrożono dwa niezależne środowiska – docelowe i testowe;
- przygotowano narzędzie użytkowe umożliwiające import danych istniejących w relacyjnej bazie danych;
- przeniesiono dotychczasowe dane o użytkownikach portali do środowiska testowego przy jednoczesnej ich anonimizacji ze względu na konieczność ochrony danych osobowych;
- przygotowano zestaw testów obciążeniowych;
- przeprowadzono testy mające na celu sprawdzenie wydajności opracowywanego rozwiązania.

3 Etap badawczy

Ze względu na niezadowalające wyniki przeprowadzonych testów obciążeniowych, przystąpiono do etapu o charakterze badawczym, którego celem miało być wykazanie przyczyn istniejących problemów.

Przeprowadzone testy obciążeniowe miały na celu symulację działania wielu użytkowników wykonujących typowe czynności: autoryzacja, przeszukiwanie bazy danych ze względu na różne kryteria, dodawanie i usuwanie użytkowników.

Testy pozwoliły zidentyfikować problemy wydajnościowe związane z najczęstszymi typami zapytań kierowanych do aplikacji przez użytkowników portalu. W kolejnych punktach zaprezentowano napotkane problemy oraz sposób ich rozwiązania.

3.1 Iteracja elementów wyniku

Zastosowany pakiet *Netscape Directory SDK for Java 4.0* [4] pozwala na przeszukiwanie katalogu LDAP w trybie synchronicznym i asynchronicznym. W trybie asynchronicznym (domyślnym)

rezultat przeszukiwania zwracany jest w paczkach o określonym rozmiarze. W trybie synchronicznym rezultat jest zwracany w całości. W obu przypadkach wynikiem jest iterator pozwalający na pobieranie kolejnych elementów wyniku. W przypadku jednak rozwiązania asynchronicznego:

- czas iteracji po elementach wyniku jest dłuższy, bowiem w razie potrzeby musi być pobrana kolejna paczka elementów;
- rozmiar wyniku nie jest znany do czasu pobrania wszystkich jego elementów (metoda badająca ten rozmiar zwraca błędną wartość);
- porzucenie wyniku bez iteracji po wszystkich elementach nie zwalnia połączenia do LDAP-a.

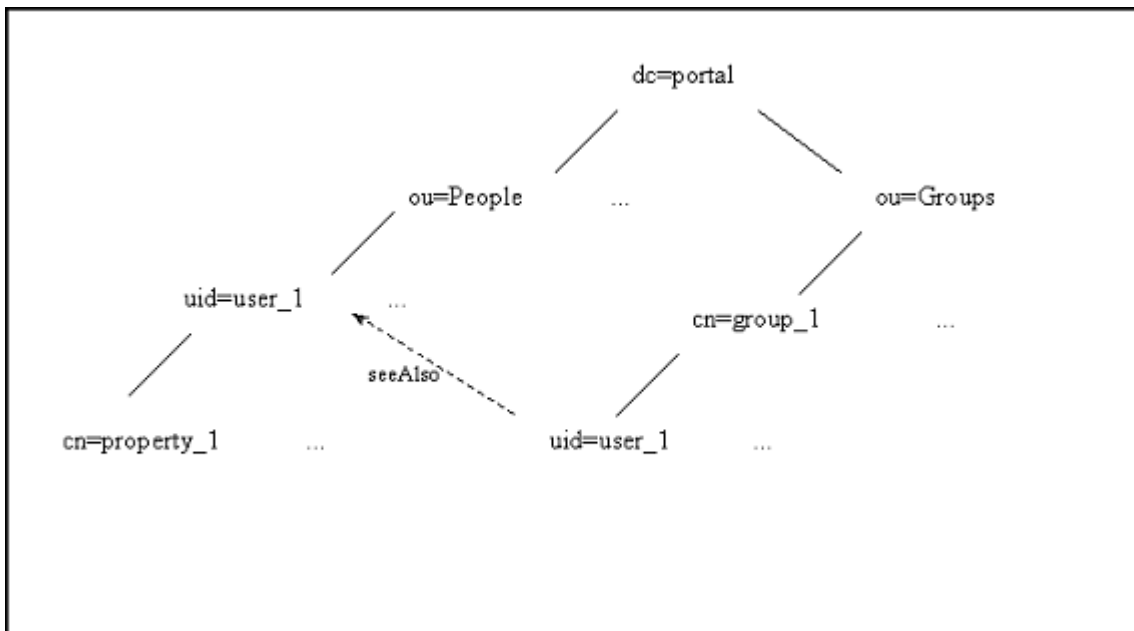
Natomiast w przypadku rozwiązania synchronicznego:

- czas iteracji po elementach wyniku jest dużo krótszy;
- metoda badająca ten rozmiar zwraca poprawną wartość;
- iterację wyniku można przerwać w dowolnej chwili a połączenie do LDAP-a zostanie poprawnie zwolnione;
- czas oczekiwania na wynik jest odpowiednio dłuższy (równy sumie czasów oczekiwania na kolejne paczki przy rozwiązaniu asynchronicznym).

Ze względu na mnogość zapytań mało selektywnych i jednocześnie nakładających ograniczenie na liczbę zwracanych elementów (tj. takich, przy których iterację wyniku można przerwać, gdy zbierze się odpowiednią ilość elementów, skracając czas wykonania) zdecydowano się przyjąć rozwiązanie synchroniczne ze wszystkimi tego konsekwencjami.

3.2 Złożone zapytania

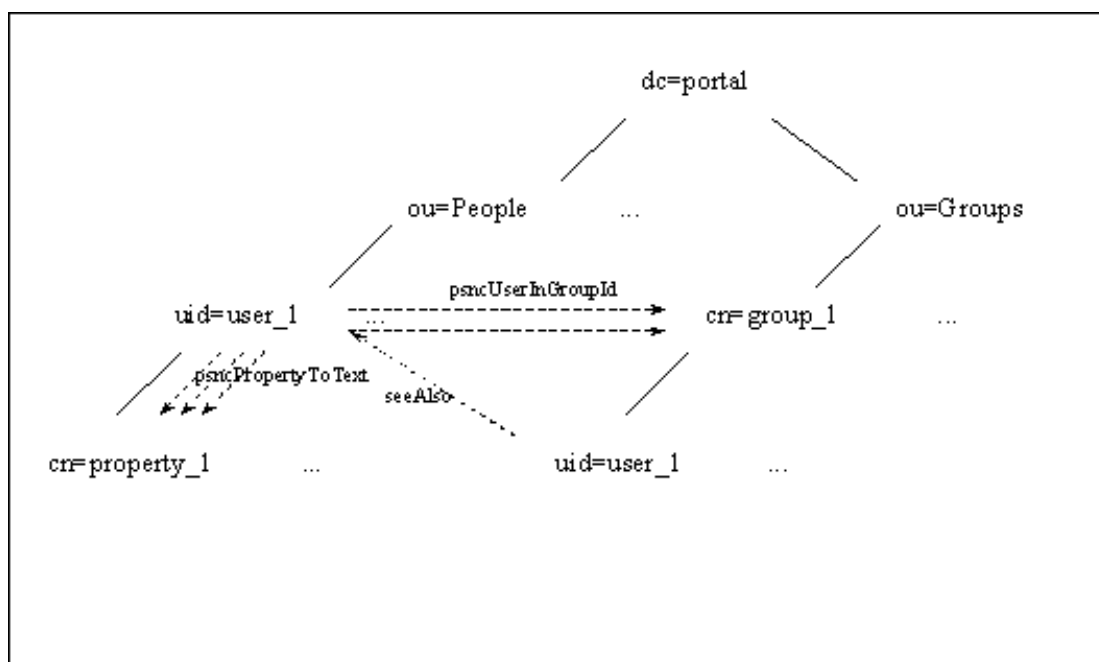
Kierowane do modułu zapytania często dotyczą elementów, których dwa lub więcej elementów podrzędnych spełnia pewne warunki. Brak możliwości odnalezienia takich elementów przy pomocy jednego odwołania do katalogu, wymuszały wykonywanie w takim przypadku wielu wyszukiwań i złożone porównywanie wyników. Zdecydowano się zmienić strukturę katalogu użytkowników, by uniknąć takiej potrzeby dla najczęściej występujących typów zapytań.



Rysunek 1: Pierwotna struktura katalogu użytkowników

Zmiany te to wprowadzenie dodatkowych atrybutów użytkownika portalu:

- wielowartościowego atrybutu *psncUserInGroupId* wskazującego grupy, do których należy użytkownik;
- wielowartościowego atrybutu *psncPropertyToText*, którego wartości to uporządkowane pary: nazwa własności, wartość własności w formacie: [nazwa] [wartość].



Rysunek 2: Zmieniona struktura katalogu użytkowników

Rozwiązanie to pozwoliło znacząco skrócić czas wykonywania pewnych typów zapytań wymagających uprzednio wielokrotnego przeszukiwania katalogu i scalania wyników.

Informacja o przynależności do grup użytkowników oraz o posiadanych własnościach jest jednak zdublowana. Przyjęte rozwiązanie wprowadza więc konieczność synchronizacji danych w różnych obiektach w czasie operacji wstawiania, usuwania i zmiany informacji o użytkownikach, co stwarza niebezpieczeństwo utraty spójności tych danych. Niebezpieczeństwo to potęguje brak transakcyjnego charakteru wykonywania operacji w bazie LDAP.

Ze względów wydajnościowych oraz z uwagi na stosunkowo mały udział operacji modyfikujących stan bazy danych użytkowników zdecydowano się przyjąć rozwiązanie ponosząc ryzyko z nim związane. Opracowano również narzędzie do badania spójności danych.

3.3 Sortowanie wyniku

Znacząca część czasu wykonania zapytań przypadła na sortowanie wyniku. W pierwotnej wersji narzędzia do sortowania używano metody dostarczanej przez zastosowane API. Metoda okazała się jednak wadliwa – proces sortowania wymagał nieracjonalnie dużo zasobów, trwał długo i powodował błędy stosu przy próbie zastosowania do większych zbiorów danych.

Zaimplementowano i wykorzystano własną metodę sortowania bazującą na metodzie porównującej dwa elementy wyniku dostarczanej przez API. Nowa metoda daje znacząco lepsze rezultaty i pozwala skrócić czas wykonania standardowych zapytań wymagających sortowania wyniku.

3.4 Indeksy

W celu zoptymalizowania operacji przeszukiwania bazy zostały założone indeksy na dodatkowe oprócz domyślnie indeksowanych atrybutów (cn, givenName, mail, seeAlso, sn, uid – dotyczy *Sun ONE Directory Server*) :

- psncUserId – zawiera identyfikator liczbowy użytkownika
- psncPropertyToText (opisany w punkcie 3.2).

4 Podsumowanie

Pakiet portalowy *Joshua* został wyposażony w zestaw narzędzi umożliwiających korzystanie z katalogu LDAP do przechowywania informacji o użytkownikach. Otwiera to możliwość współdzielenia tej informacji przez grupę portali rozwijanych w oparciu o ten pakiet. Zbadano strukturę typowych zapytań kierowanych przez aplikację portalu do modułu obsługi danych użytkowników. Wynikiem badań było zaproponowanie zmodyfikowanej struktury katalogu pozwalającej na zoptymalizowanie czasu wykonania typowych zapytań.

Migracja danych z relacyjnej bazy danych do katalogu LDAP wymaga projektowania struktur danych w bazie katalogowej ze szczególnym uwzględnieniem :

- charakteru danych przechowywanych w bazie relacyjnej;
- analizy typowych zapytań wykonywanych na tych danych;
- optymalizacji proponowanego schematu pod kątem wydajnego przeszukiwania składowanych danych (w tym wprowadzenia redundancji danych);
- konieczności zapewnienia atomowości operacji złożonych na poziomie aplikacji;
- zaproponowania utworzenia indeksów na wybranych atrybutach.

Bibliografia

- [1] Chochulski J., Czyrnek M.: [Projekt przeniesienia informacji o użytkownikach aplikacji portalowej z relacyjnej bazy danych \(Oracle\) do bazy LDAP-owej](#)
- [2] Chochulski J., Czyrnek M, Glapski J.: [Projekt aplikacji do przenoszenia danych z relacyjnej bazy danych \(Oracle\)](#)
- [3] Ustawa z dnia 29 sierpnia 1997 o Ochronie danych osobowych (Dz.U. 1997 Nr 133 poz. 883)
<http://isip.sejm.gov.pl/PRAWO.nsf/Main/WDU19971330883?OpenDocument>
- [4] Netscape Directory SDK for Java
<http://developer.netscape.com/tech/directory/index.html>