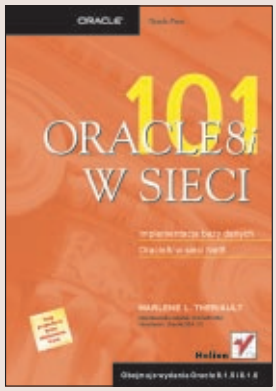
Oracle8i w sieci



Autor: Marlene L. Theriault
Tłumaczenie: Przemysław Szeremiota
ISBN: 83-7197-647-X
Tytuł oryginału: Oracle8i Networking 101
Format: B5, stron: 488

Książka „Oracle 8i w sieci" omawia wzajemne oddziaływanie poszczególnych elementów sieci komputerowej, wymagane oprogramowanie i sprzęt oraz sposób, w jaki poszczególne składniki interfejsu sieciowego komunikują się ze sobą umożliwiając łączność pomiędzy bazami danych a komputerami. Prezentowane procedury konfiguracyjne ilustrowane zrzutami ekranowymi stanowią szczegółową demonstrację sposobu konfiguracji sieci Oracle8i. Omówienie obejmuje również kwestie związane z protokołami stosowanymi w sieci Internet oraz z metodami szyfrowania transmisji.

- Poznaj sprzętowo-programowe wymagania sieci Oracle8i.
- Skonfiguruj sieć Net8 jako środek komunikacji pomiędzy serwerami a aplikacjami klientów.
- Skonfiguruj serwery Oracle Names.
- Wykorzystaj narzędzia konfiguracyjne, takie jak Net8 Assistant, Net8 Configuration Assistant i Connection Manager.
- Zwiększ wydajność sieci rozległej za pomocą serwerów wielowątkowych i technik równoważenia obciążenia.
- Korzystaj z serwerów Oracle WebDB, internet Application Server (iAS) i serwera katalogowego Internet Directory Server.
- Zapewnij bezpieczeństwo korzystania z sieci i rozwiązuj pojawiające się problemy.

Ta napisana przez administratora bazy danych Oracle i autoryzowana przez Korporację Oracle książka jest wyśmienitym podręcznikiem, zawierającym wszystkie informacje niezbędne do zaprojektowania i wdrożenia bazy danych Oracle8i w środowisku sieciowym.

Wydawnictwo Helion
ul. Chopina 6
44-100 Gliwice
tel. (32)230-98-63
e-mail: helion@helion.pl

Spis treści

Rozdział 5.

Oracle Internet Directory

Często jestem zmuszana do kontaktowania się z lokalnym biurem SSA (Social Security Administration). Sięgam wtedy po książkę telefoniczną i sprawdzam strony urzędów (oznaczone kolorem niebieskim). Wpisy w książce telefonicznej są ułożone są w kolejności alfabetycznej, co pozwala na szybkie odnalezienie numeru biura SSA oraz adresu informacyjnej strony internetowej.

Na podstawie tego przykładu można przypuszczać, że katalog (podobnie jak książka telefoniczna) jest mechanizmem lub obiektem przechowującym informacje, które dotyczą jednego lub więcej tematów. Katalogi są tworzone z określoną logiką, przechowując na przykład wpisy w kolejności alfabetycznej i umożliwiają efektywne wyszukiwanie informacji.

Pamiętając o definicji określającej katalog jako mechanizm lub obiekt przechowujący informacje o jednym lub kilku obiektach, można się rozejrzeć i zlokalizować w swoim otoczeniu jakieś katalogi. Jeżeli Czytelnik znalazł słownik, dysponuje katalogiem słów. Książka kucharska jest katalogiem przepisów. Przysłana pocztą oferta pobliskiego sklepu komputerowego może być potraktowana jako katalog rzeczy, które można kupić w tym sklepie. Podobnie jest w przypadku programu telewizyjnego. Kolejnym przykładem katalogu, przechowującego olbrzymią ilość informacji i znajdującego się w niemal każdym domu, jest encyklopedia.

Czytelnik zapewne zastanawia się, co katalogi mają wspólnego z pracą w sieci, poza wykorzystywaniem ich do przechowywania plików konfiguracyjnych Net8. Otóż w niniejszym rozdziale omówię nowy składnik Oracle8i o nazwie Oracle Internet Directory (OID). Technologia ta została zaprojektowana do współpracy z katalogiem Lightweight Directory Access Protocol (LDAP). Brzmi to nieco onieśmielająco, ale nie należy się tym przejmować. Za chwilę opowiem wszystko o działaniu LDAP i o Oracle Internet Directory, o ich zastosowaniach i konfiguracji. Oracle Internet Directory jest bardzo złożonym narzędziem, a więc napisanie wszystkiego o nim w jednym, krótkim rozdziale jest niemożliwe. Dlatego też ograniczę się do informacji pomocnych podczas rozpoczynania pracy i umożliwiających lepsze zrozumienie zastosowanej w nim technologii. W razie konieczności skorzystania z bardziej szczegółowych informacji

polecam lekturę opracowanego przez Oracle podręcznika „Oracle Internet Directory Administration Guide, Release 2.0.6", pozycja numer A77230-01.

Zapraszam zatem na wędrówkę po katalogach elektronicznych.

Różne oblicza katalogów

Przed rozpoczęciem szczegółowego omawiania tematu chciałabym wyjaśnić jedną kwestię. Stosując w tym rozdziale słowo „katalog" mam na myśli wyspecjalizowaną, elektroniczną bazę danych, w której informacje szczególnego rodzaju są przechowywane w ściśle określony sposób. Nie chodzi mi o ten rodzaj katalogu, w którym są przechowywane pliki. Pamiętanie o tym może stanowić nie lada trudność. Człowiek, który dłużej pracuje przy komputerze, przyzwyczaja się, że organizacja i rozmieszczenie plików powinno ułatwiać uzyskanie dostępu do nich. Pliki mogą być przechowywane w wielu różnych zbiorach nazywanych katalogami. Każdy katalog systemu plików posiada nazwę, która wprost mówi o jego zawartości (lub przynajmniej daje wskazówkę dotyczącą rodzaju tej zawartości).

Czytelnik zapewne już wie, czym jest katalog systemu plików. Jednak tematem tego rozdziału jest inny rodzaj katalogu. Jest to rodzaj wyspecjalizowanej bazy danych, w której są przechowywane niewielkie porcje informacji. Przykładowo, w przypadku aplikacji obsługującej pocztę jest to lista znajomych i ich adresów e-mail. W niektórych aplikacjach tego rodzaju istnieje dostęp do książki adresowej, w której wpisuje się kilka pierwszych liter imienia czy nazwiska, a mechanizm wyszukujący lokalizuje wszystkie pasujące pozycje. Książka ta jest jednak dostępna tylko za pośrednictwem obsługującej je aplikacji.

Taki rodzaj katalogu znakomicie spełnia rolę źródła informacji wymagających zarządzania. Jednym z najpopularniejszych zastosowań takich katalogów jest przechowywanie nazw komputerów, ich adresów IP oraz innych informacji istotnych w procesie odwzorowywania nazw usług sieciowych. Innym zastosowaniem jest przechowywanie list kontroli dostępu uprawniających poszczególnych użytkowników do dostępu do danego komputera czy usługi.

Katalogowa baza danych

Po przeanalizowaniu powyższych informacji warto je odnieść do relacyjnej bazy danych. Jest to przecież mechanizm umożliwiający przechowywanie i organizację informacji. Jest to także zestaw katalogów i rozmieszczonych w nich plików, które przechowują dane i metadane, do których odwołuje się mechanizm bazy danych podczas manipulacji tymi danymi. Wszystkie elektroniczne bazy danych, jakie miałam okazję poznać, składają się z plików rozmieszczonych w katalogach. Trzeba jednak cały czas pamiętać, że nie mówię tu o katalogach w sensie struktury systemu plików. Mówię o relacyjnych bazach danych i katalogowych bazach danych. Należy wobec tego określić różnice pomiędzy tymi dwoma rodzajami baz danych.

Katalogowe i relacyjne bazy danych

Podstawowa różnica między omawianymi strukturami wynika ze sposobu korzystania z relacyjnej i katalogowej bazy danych.

Najpierw przypomnę sposób korzystania z relacyjnej bazy danych. Przykładem niech będzie aplikacja bazy danych w miejscowej uczelni. Taka baza danych składa się z tabel zawierających informacje o wykładach, dane personalne studentów, informacje o przedmiotach, na jakie uczęszczają poszczególni studenci, oceny z egzaminów. W skład takiej bazy danych wchodzą również formularze rejestracyjne. Student może przeglądać formularz rejestracyjny i zapisać się za jego pomocą na jeden lub kilka przedmiotów lub zmodyfikować swoje dane personalne. Może również sprawdzić rozkład zajęć — być może posiada nawet uprawnienia do przeglądania swoich ocen. Wykładowca może wprowadzać oceny poszczególnych studentów. Dzięki możliwościom relacyjnej bazy danych może odszukać oceny studenta na podstawie jego danych personalnych. Pracownik administracyjny może zaś przetwarzać formularze rejestracyjne studentów oraz wprowadzać, usuwać czy aktualizować informacje o przedmiotach.

Przeprowadzane operacje polegają zatem na wstawianiu, aktualizacji i — rzadziej — usuwaniu danych za pomocą kilku zapytań. Wprawdzie wyszukiwanie informacji w hurtowni danych nie polega na takich manipulacjach danymi, ale zdecydowana większość operacji realizowanych przez bazę danych polega na przechowywaniu, pobieraniu, wstawianiu, aktualizacji i usuwaniu dużych ilości informacji. Tak więc można śmiało stwierdzić, że relacyjna baza danych jest bardzo obciążona obsługą zapisu rozmaitych rodzajów informacji. Poza tym mechanizm relacyjnej bazy danych poszukuje potrzebne informacje w konkretnym miejscu, w określonym pliku przechowywanym w zdefiniowanym komputerze. Położenie danych relacyjnej bazy danych jest zapisane w słowniku bazy danych.

Warto przeanalizować strukturę katalogowej bazy danych. Zazwyczaj praca z katalogową bazą danych wymaga jej początkowego wypełnienia danymi, ale po załadowaniu danych do bazy większość operacji polega na wykonywaniu zapytań i wyszukiwaniu drobnych porcji informacji. Informacje te są przechowywane w parach klucz-wartość, przykład takiej pary stanowią nazwisko znajomego i adres jego poczty elektronicznej. Katalogowa baza danych raczej służy do odczytywania danych i nadaje się do obsługi stosunkowo prostych transakcji. W katalogowej bazie danych nie przechowuje się (lub przechowuje się niewielkie ilości) informacji o związkach pomiędzy danymi. W celu pobierania informacji z katalogowej bazy danych zazwyczaj wykorzystuje się aplikację serwera usług katalogowych. Udostępnia on przechowywane w katalogu informacje wszystkim aplikacjom klientów, które wykonują zapytania do serwera usług katalogowych.

Serwery usług katalogowych

Nareszcie dotarłam do sedna tego rozdziału. Jestem pewna, że dość już czasu poświęciłam na wstęp. Dokonam teraz ogólnego przeglądu serwerów usług katalogowych, po czym omówię implementację Oracle serwera usług katalogowych LDAP. Nawiasem mówiąc, gdy pisałam ten rozdział, korporacja Oracle określiła tę nowo powstającą technologię mianem Oracle Internet Directory. Mówiąc ogólnie o serwerach usług

katalogowych będę więc stosowała termin „serwer usług katalogowych LDAP", a termin „Oracle Internet Directory" zarezerwuję dla wspomnianej implementacji serwera usług katalogowych, opracowanej przez korporację Oracle.

Omawiając cechy relacyjnych baz danych wspomniałam, że mechanizm bazy danych może określić położenie wyszukiwanych danych. Natomiast informacje serwera katalogowego są niezależne od komputera. Cóż to oznacza dla użytkownika?

Czytelnik zapewne pamięta serwer Oracle Names, który omawiałam w rozdziale 4. Podczas konfigurowania serwera Oracle Names istnieje możliwość określenia, że serwer będzie przechowywał informacje dotyczące odwzorowania nazw i adresów w pamięci podręcznej. Przy takim sposobie przechowywania danych wszelkie nowo zarejestrowane nazwy są przekazywane do wszystkich pozostałych serwerów Oracle Names, uruchomionych w danym w regionie. Struktura aplikacji serwera usług katalogowych powoduje, że w całym środowisku są dostępne te same informacje, niezależnie od tego, z którego serwera usług katalogowych są pobierane informacje. W przypadku gdy żądanie klienta nie może zostać obsłużone lokalnie, serwer wyszukuje żądane informacje lub wskazuje klientowi miejsce, w którym one się znajdują. Oczywiście, z punktu widzenia klienta jest to proces przeźroczysty.

Podobnie jak serwer Oracle Names, tak i serwer usług katalogowych LDAP umożliwia translację żądań lokalizacji informacji. Jednak w przeciwieństwie do serwera Oracle Names, serwer usług katalogowych LDAP umożliwia translację obiektów spoza rodziny Oracle. Serwer usług katalogowych LDAP umożliwia komunikowanie się użytkowników, wspólne wykorzystywanie aplikacji i zasobów przez komputery, sieci, a nawet państwa.

Czytelnik zapewne potrafi wyobrazić sobie, że spędza w pracy mnóstwo czasu dostosowując środowisko do osobistych preferencji. Pulpit został już dokładnie dostosowany do osobistych upodobań, a kolory okien odpowiadają jego temperamentowi. Profil użytkownika jest przechowywany lokalnie w komputerze PC. Jednak w razie łączenia się z siecią firmową z domu ten dostosowany profil jest niedostępny. Aby w ten sam sposób skonfigurować środowisko pracy na komputerze domowym, trzeba by ustawiać wszystko od nowa. Czy nie byłoby wspaniale posiadać profil użytkownika, który byłby dostępny z dowolnego miejsca na świecie? Katalog udostępnia środki, za pomocą których użytkownicy mogą zawsze mieć dostęp do swoich prywatnych ustawień środowiskowych.

Serwery usług katalogowych LDAP

Każdy kierowca wie, że podczas jazdy musi dostosować się do wielu przepisów i zasad ruchu. Przepisy te wyznaczają czynności, jakie w danych okolicznościach należy podjąć, aby nie popełnić wykroczenia. Przepisy określają, przykładowo, maksymalną prędkość jazdy w pobliżu szkół lub wymuszają zatrzymanie pojazdu przed znakiem stop. Przepisy mogą zobowiązywać kierowcę do zatrzymania pojazdu na czerwonym świetle na skrzyżowaniu, nawet jeżeli ma on zamiar skręcić w prawo. Wszystkie te zasady i regulacje zostały określone w celu zapewnienia bezpieczeństwa użytkownikom dróg.

W rozdziale 1. i 2. pisałam o rozmaitych protokołach, na podstawie których korporacja Oracle opracowuje swoje technologie sieciowe. Protokoły te stanowią zbiór reguł definiujących sposób przesyłania danych w sieci komputerowej. Podobnie jak przepisy ruchu drogowego, protokoły są uzgodnionymi zasadami transmisji, chroniącymi ruch sieciowy przed kolizjami i zatorami.

Jedną z zalet stosowania serwera usług katalogowych LDAP jest to, że dysponuje się dzięki niemu scentralizowaną lokalizacją, w której można przechowywać informacje różnego typu. Serwer usług katalogowych może służyć jako centralne repozytorium wszystkich informacji o komponentach sieciowych, stosowanych politykach bezpieczeństwa użytkowników i działów, a nawet danych uwierzytelniających użytkowników, łącznie z ich hasłami. Dzięki takiemu pojedynczemu repozytorium ewentualne zmiany są dokonywane w jednym miejscu, a nie w potencjalnie wielkiej liczbie plików. Znika też konieczność utrzymywania pliku *tnsnames.ora* na komputerach klientów i serwerów. Katalog przechowuje nazwy pozostałych usług, a więc jedna konfiguracja może umożliwiać nawiązywanie połączenia z serwisami działającymi na podstawie różnych protokołów.

Korzystanie z serwera usług katalogowych LDAP ma też swoje wady. Jeżeli jedyny działający w sieci serwer LDAP ulegnie awarii, żądania klientów przestaną być obsługiwane. Trudno też przewidzieć konsekwencje sytuacji, w której uruchamiana aplikacja nie może nawiązać połączenia lub jej składniki są rozproszone w taki sposób, że bez pomocy serwera usług katalogowych nie są dostępne.

Inną niedogodnością jest fakt, że wszystkie klienty muszą mieć dostęp do katalogu. Może to stanowić potencjalną lukę w systemie zabezpieczeń systemu. Udostępnienie użytkownikom serwera katalogu LDAP jest pewnego rodzaju otwarciem furtki dla hakerów, którzy dzięki niemu mogą potencjalnie uzyskać informacje o wszystkich komponentach sieciowych, a być może i dostęp do danych o strategicznym znaczeniu dla przedsiębiorstwa. Ostrzeżenia te mają uświadomić Czytelnikowi, że zaprojektowanie i konfiguracja serwera usług katalogowych może powodować pewne problemy. Wielu z tych problemów można uniknąć dzięki starannemu zaplanowaniu struktury katalogu i uwzględnieniu na tym etapie ewentualnych trudności.

Przykładowo, aby zabezpieczyć się na wypadek awarii serwera katalogu, można wykorzystać lokalne bufory przechowujące dane, które pobrano ostatnio z serwera usług katalogowych. Możliwe, że zawartość pamięci podręcznej nie uwzględnia najnowszych danych zaktualizowanych w katalogu, ale rozwiązanie to zapewnia przynajmniej dostęp do potrzebnych informacji nawet w przypadku niedostępności serwera katalogu.

Czytelnik otrzymał już sporą dawkę ogólnych informacji o serwerach usług katalogowych LDAP. Należy jeszcze omówić ich budowę. Zachęcam także do zapoznania się z odrobiną historii LADP i ze swojego rodzaju „przepisami ruchu", będącymi podstawą specyfikacji tego protokołu.

Usługi katalogowe LDAP — trochę historii

Protokół uproszczonego dostępu do katalogu (*Lightweight Directory Access Protocol*, LDAP) został opracowany przez grupę Internet Engineering Task Force (IETF) i jest otwartym standardem internetowym. Innymi osiągnięciem zespołu IETF jest opracowanie protokołów TCP/IP, DNS, SMTP, NNTP, SNMP i HTTP.

Protokół X.500, definiujący usługi katalogowe modelu OSI, uwzględnia wiele świetnych koncepcji, ale okazuje się trudny w implementacji i wdrażaniu w sieci Internet. Protokół X.500 jest po prostu zbyt skomplikowany i rozbudowany. Implementacje tego protokołu są trudne w stosowaniu i wymagają niemałych zasobów, którymi przeciętny użytkownik nie dysponuje. Protokół LDAP został opracowany jako uproszczony interfejs protokołu X.500 *Directory Access Protocol*. Zakładano, że LDAP miał udostępniać 90% funkcji X.500 i zmniejszać zapotrzebowanie na zasoby do 10%. Protokół LDAP umożliwia uproszczony dostęp do katalogu i posiada następujące właściwości:

- działa na podstawie protokołu TCP/IP i eliminuje wyższe warstwy wielowarstwowego stosu komunikacyjnego OSI, uwzględniane w protokole X.500;
- eliminuje niewykorzystywane funkcje i nadmiarowe operacje protokołu X.500, przez co znacznie go upraszcza;
- korzysta z prostych formatów ciągów połączeniowych dla poszczególnych elementów danych — ciągi połączeniowe X.500 są bardziej skomplikowane i posiadają ściśle strukturalną reprezentację;
- w porównaniu do wymagań protokołu X.500 protokół LDAP upraszcza reguły szyfrowania transmisji danych.

Pierwotnie LDAP służył jako interfejs dla protokołu X.500. Klient LDAP mógł nawiązać połączenie z serwerem LDAP, który z kolei mógł przekazać żądanie klienta do serwera X.500. Połączenia takie były obarczone narzutami czasowymi związanymi z obsługą skomplikowanego i niewygodnego serwera X.500, obsługującego w tle żądanie klienta. Z biegiem czasu model LDAP został oddzielony od protokołu X.500 i opracowano pierwsze samodzielne serwery LDAP. W modelu LDAP klient zgłasza żądanie bezpośrednio do serwera LDAP i z tego samego serwera otrzymuje natychmiastową odpowiedź. Zastosowanie modelu LDAP i wyeliminowanie czynności wymaganych w protokole X.500 powoduje, że serwery LDAP działają równie wydajnie, jak inne proste serwery internetowe, które cechują się dużym stopniem integracji ze środowiskiem Internetu.

Modele LDAP

Wyróżnia się cztery modele opisujące operacje, przechowywanie danych i sposób użycia LDAP. Modele te obejmują:

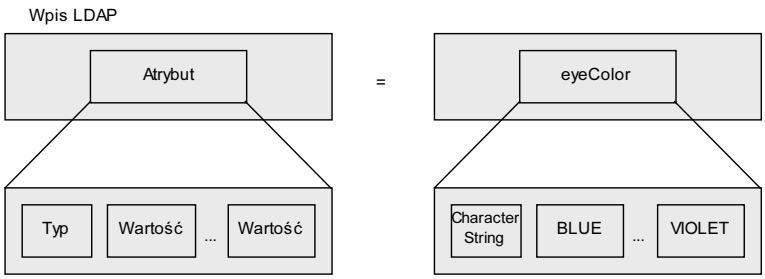
- model informacyjny, definiujący rodzaj przechowywanych informacji;
- model nazw, definiujący organizację i sposób odwoływania się do informacji katalogu LDAP;
- model funkcjonalny, opisujący możliwości przetwarzania i aktualizacji informacji oraz sposoby dostępu do nich;
- model bezpieczeństwa, definiujący sposób zabezpieczania katalogu LDAP przed nieuprawnionym dostępem.

Model informacyjny

Zastanawiam się, w jaki sposób Czytelnik opisałby samego siebie. Czy jest wysoki, a może niski, czy jest tęgi, średniej budowy, czy może szczupły? Jakiego koloru są jego oczy i włosy? Czy jest mężczyzną, czy kobietą?

Gdybym zechciała wykorzystać katalog do przechowywania opisów prezencji swojej i swoich znajomych, opis każdej z osób mógłby zajmować jeden wiersz, czyli wpis. Każdy wpis składałby się z atrybutów takich jak wzrost, waga, kolor włosów, oczu, płeć. Każdemu z atrybutów przypisano by pewne reguły (ograniczenia). Reguły te są nazywane typami. Przykładowo, atrybut definiujący kolor oczu może być ciągiem znakowym o nazwie `eyeColor`. Z każdym typem są związane wartości dopuszczalne dla danego atrybutu. Typem atrybutu `eyeColor` jest ciąg znakowy, a jego wartościami dopuszczalnymi są `BLUE`, `BROWN`, `HAZEL`, `VIOLET` itd. W ten sposób określa się model informacyjny, definiujący strukturę wpisu w katalogu LDAP. Opiszę ten model w bardziej formalny sposób.

Model informacyjny LDAP definiuje rodzaje przechowywanych informacji, kładąc nacisk na poszczególnych wpisach. Zasadniczo, wpisy dotyczą pojęć lub obiektów świata rzeczywistego, takich jak osoby, organizacje, drukarki itd. Nie jest to jednak konieczne — wpisy mogą też dotyczyć pojęć abstrakcyjnych. Powiedziałam już, że wpisy składają się z atrybutów zawierających informacje dotyczące obiektu oraz że każdy atrybut posiada typ dopuszczający jedną lub więcej wartości. Rysunek 5.1 przedstawia ogólną strukturę wpisu, z jednym atrybutem pewnego typu i jego wartościami. Obok ogólnego modelu wpisu znajduje się przykład wpisu z atrybutem `eyeColor` typu ciąg znaków i jego wartości.



Rysunek 5.1. *Model ogólny i przykład wpisu w katalogu LDAP*

Warto dokładniej omówić kwestię typu atrybutu. Aby określić rodzaj informacji, która może być przechowywana jako wartość atrybutu, należy sprawdzić składnię typu atrybutu. Typ określa również sposób zachowywania się danej wartości w czasie wyszukiwania, porównywania lub innych operacji katalogu. Przykładowy atrybut `Common Name` (nazwa), który w nomenklaturze LDAP jest określany skrótem `cn`, posiada składnię `caseIgnoreString`. Składnia ta wskazuje, że podczas porównywania wartości atrybutu będzie ignorowana wielkość liter, składających się na tę wartość, oraz że wartość ta musi być ciągiem znakowym. Z tego względu wpisy `NOWAK`, `Nowak` oraz `nowak` mają, zgodnie ze składnią, identyczne wartości. Atrybut `todayDate` posiada identyczną składnię `caseIgnoreString`, która tym razem oznacza ignorowanie wszelkich kresek i spacji w porównywanych datach. Dzięki temu data `10-20-2000` i `10202000` podczas porównywania są traktowane jako identyczne.

Na atrybuty można również nakładać pewne ograniczenia, takie jak limit długości, układ, liczbę argumentów i tym podobne. Atrybut przeznaczony do przechowywania numeru karty kredytowej może na przykład być ograniczony do przyjmowania tylko jednej wartości wejściowej, a atrybut służący do przechowywania tekstu dokumentu może być ograniczony maksymalną liczbą słów. Reguły dotyczące zawartości służą z kolei do określania wymaganych lub dozwolonych wartości atrybutów. W miejsce reguł zawartości można w każdym wpisie zastosować specjalny atrybut o nazwie `objectClass`. Atrybut `objectClass` definiuje typ wpisu, a także określa wymagane i opcjonalne atrybuty. Atrybut `objectClass` dla wpisu `OSOBA` może, przykładowo, wymagać określenia atrybutu `sn` (*surname*, nazwisko), `cn` (nazwa) i innych. Równoważnikiem takiej struktury w bazie danych jest schemat. Aby zmienić bieżący schemat bazy danych LDAP, należy dodać do wpisu nowe klasy.

Każdy z wpisów posiada specjalną klasę, nazywaną *klasą obiektu strukturalnego*, definiującą rodzaj wpisu. Klasa obiektu strukturalnego nie może być modyfikowana. Reszta klas to klasy *pomocnicze*. Klasy pomocnicze mogą być dodawane do wpisu lub usuwane z niego na podstawie obowiązujących reguł dostępu. Wersja 3. standardu LDAP uwzględnia specjalną klasę o nazwie `extensibleObject`, służącą do nadpisywania aktualnie obowiązujących reguł schematu. Czasami przykrycie obowiązujących reguł schematu nowymi regułami jest pożądane, na przykład jeśli zdefiniowanie nowej reguły schematu i uwzględnienie tej zmiany po stronie serwera i po stronie klientów mogłoby wymagać dużego nakładu pracy. W takim przypadku proste przykrycie reguły jej nową wersją oraz swobodne dodawanie i usuwanie atrybutów może być dużo wygodniejszą metodą.

Rozszerzenia LDAP 3

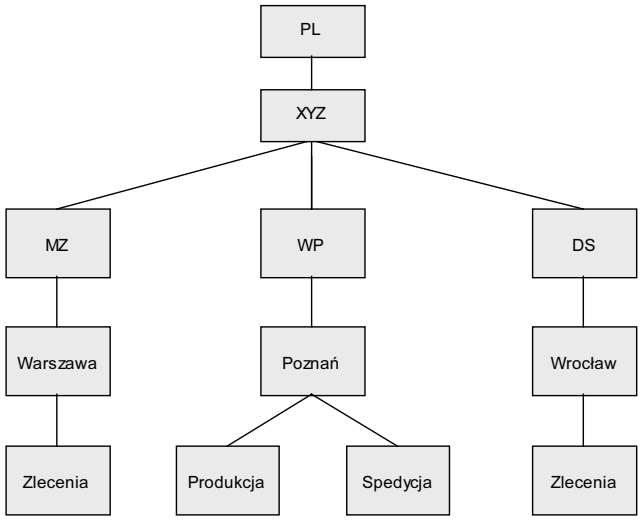
Ponieważ wspominałam o wersji 3. protokołu LDAP, chciałabym zaznaczyć, że trzecia wersja standardu LDAP została przyjęta w grudniu 1997 roku przez IETF, jako standard obowiązujący w Internecie. Nowe standardy określają szereg rozszerzeń, z których korzysta Oracle. Standardy te umożliwiły korporacji Oracle implementację w serwerze Oracle Internet Directory następujących funkcji:

- obsługa znaków diakrytycznych wszystkich języków świata;
- globalne rozmieszczenie drzewa *Directory Information Tree* pomiędzy wieloma serwerami LDAP za pomocą mechanizmu referałów (mechanizm ten skrótowo objaśniono w podrozdziale „Referały LDAP”);
- implementacja i obsługa standardowych protokołów *Simple Authentication and Security Layer* (SASL) oraz *Transport Layer Security* (TLS), umożliwiających zastosowanie wszechstronnej platformy zabezpieczania danych LDAP;
- umożliwienie producentom oprogramowania rozszerzania operacji LDAP za pomocą mechanizmu o nazwie *controls*;
- publikacje informacji przydatnych innym serwerom LDAP i klientom.

Model nazw

W rozdziale 2. omawiałam struktury hierarchiczne. Czytelnik zapewne pamięta wykres hierarchii kierownictwa przedsiębiorstwa XYZ (rysunek 5.2). Rysunek 5.2 wygląda podobnie do rysunku 2.5 prezentowanego w rozdziale 2., ale różni się od niego kilkoma szczegółami. Dodano bowiem tu element oznaczony etykietą „XYZ".

Rysunek 5.2. *Wykres hierarchii kierownictwa polskiego oddziału przedsiębiorstwa XYZ*



Mimo że nie jest to wymagane przez protokół, wpisy modelu nazw LDAP są zazwyczaj wyświetlane w postaci drzewa, odzwierciedlającego strukturę geograficzną lub organizacyjną. Poszczególne wpisy są oznaczone nazwami, zgodnie z ich pozycją w hierarchii, a każdy wpis posiada nazwę wyróżniającą (*Distinguished Name*, DN). Każdy komponent nazwy jest określany jako względna nazwa wyróżniająca (*Relative Distinguished Name*, RDN). Nazwy RDN mogą składać się z jednego lub z kilku atrybutów wpisu.

Aby ułatwić sobie zrozumienie modelu nazw, można przeanalizować strukturę systemu plików Windows NT lub UNIX. RDN może być traktowany jak nazwa pliku systemu plików. Nazwa DN jest zaś w tej analogii w pełni kwalifikowaną ścieżką dostępu do tego pliku. Zaproponuję teraz, aby Czytelnik przeanalizował następującą nazwę pliku: *D:/Ora816/Oracle/Network/Admin/trace_010500.trc* i odpowiedział na pytanie, która część tej nazwy jest nazwą DN, a która RDN? Warto przemyśleć swoją odpowiedź, gdy będę w dalszym ciągu omawiała model nazw LDAP.

Dwa pliki we wspólnym katalogu nie mogą posiadać takich samych nazw — dwa wpisy o tym samym wpisie nadrzędnym w katalogu LDAP też muszą posiadać różne nazwy `cn`. Węzły końcowe i węzły wewnętrzne struktury LDAP mogą przechowywać informacje. Termin przestrzeń nazw dotyczy kombinacji lokacji tworzących kwalifikowana ścieżkę dostępu do szukanej informacji. W strukturze systemu plików systemu operacyjnego przestrzeń nazw jest zakorzeniona w najobszerniejszym elemencie i schodzi w dół, aż do nazwy pliku. Zgodnie z powyższym, ścieżka *D:/Ora81/network/admin* jest przestrzenią nazw, zakorzenioną w elemencie symbolizującym dysk D: i zagłębiającą się aż do plików umieszczonych w katalogu *admin*. W strukturze LDAP przestrzeń

nazw jest zakorzeniona w elemencie najmniej obszernym i zawiera nazwy elementów nadrzędnych, aż do korzenia katalogu. Z tego względu wyszukiwanie wpisu rozpoczyna się zawsze od elementu `cn` i przechodzi do korzenia przestrzeni nazw LDAP. Zanim zaprezentuję przykład modelu nazw, muszę wspomnieć o wykorzystywanych w nim separatorach.

W przytoczonej powyżej ścieżce dostępu do pliku separatorami były znaki ukośnika (/) lub lewego ukośnika (\), w zależności od systemu operacyjnego. W przypadku LDAP składniki nazw są oddzielane za pomocą przecinka (,). Wpis LDAP mógłby więc mieć nazwę `cn=Zlecenia,u=Warszawa,st=MZ,o=XYZ,c=PL`, w której nazwą elementu jest `Zlecenia`, jednostką (*u — unit*, z ang. jednostka) jest Warszawa, województwem (*st — state*) jest woj. mazowieckie, organizacją jest przedsiębiorstwo XYZ, a krajem (*c — country*) jest Polska. W przykładzie tym można wyróżnić nazwy względne (RDN): `cn=Zlecenia`, `u=Warszawa`, `st=MP`, `o=XYZ` oraz nazwę DN `o=XYZ,c=PL`. Przy okazji, w prezentowanej powyżej nazwie pliku nazwą RDN byłby ciąg *trace_010500.trc*, zaś nazwą DN ciąg *D:\Ora816\Network\Admin*.

Właśnie zdałam sobie sprawę z tego, że odpowiedź ta może być nieco myląca — należy przeanalizować ją dokładniej. W przypadku nazwy pliku cała ścieżka dostępu mogłaby być traktowana jak nazwa DN wpisu LDAP. Przy założeniu, że istnieje w LDAP jednostka o nazwie `pathname` (ścieżka dostępu do pliku) określana skrótem `pn`, zgodnie z modelem nazw LDAP cała ścieżka dostępu byłaby pojedynczym składnikiem nazwy wpisu: `pn=D:\Ora816\Network\Admin`. Właściwą nazwą pliku byłby element `cn`: `cn=trace_010500.trc`. W razie podzielenia składowej `pn` na mniejsze elementy, każdy z elementów poniżej korzenia byłby nazwą względną wpisu (RDN), a ciąg wszystkich elementów nazwą byłby nazwą wyróżniającą wpisu (DN). Istotą modelu nazw jest zapewnienie niepowtarzalności nazw poszczególnych elementów, co ma umożliwiać szybkie i dokładne wyszukiwanie informacji przez serwer LDAP.

Nazwa wyróżniająca DN jest w istocie sekwencją nazw względnych (RDN), rozdzielonych przecinkami (`,`) lub średnikami (`;`). Każda składowa RDN jest zbiorem par atrybut-wartość, oddzielonych znakiem plus (+). Jeśli wartość atrybutu zawiera w sobie znak separujący, wartość ta musi zostać ujęta w znaki cudzysłowu (`"`), ewentualnie znak separatora należy poprzedzić znakiem sterującym. Znakiem sterującym jest lewy ukośnik (\). Czasem zdarza się, że wartość atrybutu zawiera znak cudzysłowu lub lewego ukośnika. W takich przypadkach znaki te muszą być poprzedzane znakiem sterującym. Przykładowo, aby do atrybutu `cn` przypisać wartość `http:\\mojastronawww.com.pl`, należy napisać: `cn=http:\\\\mojastronawww.com.pl`. Każdy znak lewego ukośnika w wartości atrybutu musi zostać poprzedzony jednym znakiem sterującym — znakiem lewego ukośnika. Takie konwencje nazewnicze mogą okazać się dość skomplikowane podczas zastosowania, dlatego też warto przyjąć regułę unikania wielowartościowych składowych RDN i stosowanie przecinka jako separatora.

Oczywiście, poza zaprezentowanym formatem nazwy DN istnieją jeszcze inne formaty. Tym niemniej niniejsza prezentacja powinna dostarczyć informacji wystarczających do zapoznania się z implementacją serwera katalogowego LDAP. Dalsze szczegóły implementacji serwera katalogowego LDAP zostaną przedstawione w dalszej części niniejszego rozdziału.

Model funkcjonalny

Po zapoznaniu się z rodzajami informacji, jakie mogą być przechowywane w katalogu i po zaznajomieniu się ze stosowaną w nim konwencją nazewnictwa, należy poznać operacje umożliwiające dostęp do danych zgromadzonych w katalogu LDAP. Operacji tych jest 9 i można je podzielić na trzy kategorie:

- zapytania: wyszukiwanie, porównywanie;
- aktualizacje: dodawanie, usuwanie, modyfikacje, modyfikacje nazw RDN;
- uwierzytelnianie: wiązanie, uwalnianie i rezygnacja.

Operacje zapytania służą do przeszukiwania struktury katalogu LDAP i wyszukiwania informacji. Dzięki zastosowaniu kryteriów selekcji (filtru wyszukiwania) operacja wyszukiwania umożliwia wybieranie informacji z określonych obszarów drzewa katalogu. Wynikiem wyszukiwania może być zwrócenie zbioru atrybutów (z ich wartościami lub bez nich) każdego wpisu, pasującego do zbioru kryteriów (filtru). Klient może określić dopuszczalny czas oczekiwania na rezultat wyszukiwania, akceptowalny rozmiar lub liczbę wpisów.

Jak wskazuje nazwa, operacje aktualizacji umożliwiają dodawanie, modyfikacje i usuwanie informacji ze struktury katalogu. Operacja modyfikacji (ang. *Modify*) służy do zmieniania wartości atrybutów istniejącego wpisu. Można też dodawać i usuwać atrybuty oraz ich wartości. Operacja usuwania pozwala na usunięcie istniejącego wpisu. Aby zmienić nazwę wpisu, należy skorzystać z operacji modyfikacji nazwy RDN (ang. *Modify RDN*).

Operacje wiązania i uwalniania mają fundamentalne znaczenie dla zapewnienia bezpieczeństwa informacji przechowywanych w katalogu. Operacja wiązania umożliwia uwierzytelnienie klienta, czyli udowodnienie jego tożsamości. Aby dokonać uwierzytelnienia, klient przedstawia nazwę DN i hasło w postaci jawnej. Ciekawe jest, że serwer nie musi uwierzytelniać się w stosunku do klienta. Jeżeli uwierzytelnianie klientów nie jest konieczne, klient może przedstawić pustą (`NULL`) nazwę DN i puste hasło. Operacja uwalniania kończy sesję katalogu. Operacja rezygnacji umożliwia anulowanie danej operacji w trakcie jej wykonywania. Jest to bardzo przydatna możliwość w sytuacji, kiedy operacja wyszukiwania zajmuje zbyt wiele czasu. Wersja 3. protokołu LDAP zapewnia szczelniejszą implementację zabezpieczeń, uwzględniając uwierzytelnianie obydwóch stron transakcji — serwer również musi udowodnić klientowi swoją tożsamość.

Model bezpieczeństwa

Z opisu operacji wiązania i uwalniania, definiowanych przez model funkcjonalny, wynika, że bezpieczeństwo transakcji opiera się na uwierzytelnianiu klientów zgłaszających żądanie dostępu do katalogu LDAP. Samo uwierzytelnianie zachodzi właśnie za pomocą operacji wiązania. Po pomyślnej identyfikacji klienta są wykorzystywane informacje list kontroli dostępu, które określają uprawnienia danego klienta do wykonywania żądanych operacji. Model LDAP nie określa formatu czy właściwości list kontroli dostępu, a więc programiści mają wolną rękę w implementacji własnych

mechanizmów kontroli, odpowiednich dla danego systemu. Jeżeli dwie różne implementacje wymagają replikacji informacji, może pojawić się problem, wynikający z niezgodności mechanizmów kontroli dostępu.

Referały LDAP

W czasach, gdy serwer LDAP stanowił interfejs do usług katalogu X.500, wewnętrzny serwer usług katalogowych miał za zadanie rozstrzyganie wszystkich zapytań i zwracanie ostatecznych wyników lub komunikatów o błędzie. Jeżeli wewnętrzny serwer usług katalogowych nie mógł obsłużyć zapytania, miał nawiązywać kontakt z innymi serwerami i wykonać zapytanie w imieniu klienta — proces ten zwany jest *łączeniem łańcuchowym*. Klient nie był informowany o fakcie nawiązywania połączeń w tym celu i o angażowaniu innego serwera do wykonania zapytania.

Okazało się, że model łańcuchowy jest mało elastyczny i przez to nieefektywny w rozproszonym środowisku Internetu. Opracowano więc nowy model, zwany *modelem referencjalnym*. W mocno rozproszonym i zróżnicowanym środowisku, takim jak Internet, referały ułatwiały wdrożenie serwerów usług katalogowych LDAP. Zasadą modelu referencjalnego jest, że jeżeli serwer usług katalogowych LDAP nie dysponuje żądanymi informacjami, może odesłać klienta do innego serwera usług katalogowych.

Teraz już Czytelnik posiada pewne ogólne informacje o historii standardu LDAP i definiowanych w nim modelach, a więc może zapoznać się ze sposobem, w jaki korporacja Oracle zaimplementowała i rozszerzyła tę technologię i wykorzystała do obsługi baz danych Oracle.

Oracle Internet Directory — przegląd

Katalog Oracle Internet Directory, zaimplementowany jako aplikacja wydania 2. Oracle8i, łączy cechy wersji 3. protokołu LDAP i możliwości serwera Oracle8i. Implementacja ta obejmuje cztery komponenty:

- serwer Oracle Directory;
- serwer replikacji danych Oracle Directory Replication;
- program Oracle Directory Manager;
- tekstowe narzędzia administracyjne i narzędzia zarządzania danymi.

Serwer Oracle Directory udziela odpowiedzi i obsługuje aktualizacje żądań klientów dotyczących osób lub zasobów. Serwer replikacji Oracle Directory Replication — jak wskazuje jego nazwa — obsługuje mechanizm replikacji danych pomiędzy serwerami usług katalogowych LDAP. Jeżeli jeden z serwerów biorących udział w replikacji jest niedostępny, użytkownik może uzyskać dostęp do danych katalogu za pośrednictwem innego serwera. Oracle Directory Manager jest narzędziem administracyjnym, wyposażonym w graficzny interfejs użytkownika. Manipulacja dużymi zbiorami danych przechowywanych w katalogu i administracja serwerem Oracle Directory jest ułatwiana przez zestaw narzędzi, obsługiwanych z wiersza poleceń.

Wpisy, atrybuty i klasy

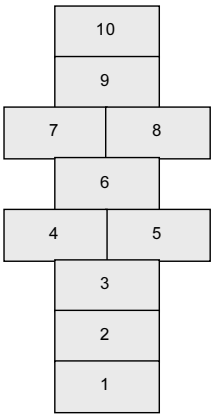
W rozdziale 1. omawiałam standardy modelu ISO-OSI i zwracałam szczególną uwagę na fakt, że w modelu tym każda z warstw protokołów jest dokładnie zdefiniowana, ale sposób jej implementacji pozostawiono w gestii producentów. Standardy serwerów usług katalogowych LDAP są podobne pod tym względem. Producenci oprogramowania mogą w swobodny sposób opracowywać aplikacje i interpretować obowiązujące standardy.

Wcześniej omawiałam rozmaite modele serwerów usług katalogowych, wspomniałam też o kompozycji serwera usług katalogowych, przyjętej przez IETF. Warto przeanalizować implementację i interpretację LDAP, wykonaną przez korporację Oracle.

Wpisy

Czytelnik zapewne zna grę w klasy. Kiedyś grałam w nią prawie codziennie. Mogę przypomnieć, jak przygotować pole do tej gry — potrzebny będzie spory kawałek kredy. Najpierw trzeba narysować na chodniku 10 prostokątów (można je przerysować z rysunku 5.3). Prostokąty te powinny być na tyle duże, aby można było stanąć w nich na jednej nodze bez deptania krawędzi. Byłabym zapomniała — ich wielkość powinna umożliwiać wykonanie obrotu na jednej nodze.

Rysunek 5.3.
Pole do gry w klasy



Aby zacząć grę, należy jeszcze znaleźć kamyk — najlepiej gładki i niezbyt mały. Kamyk powinien być na tyle duży, żeby można było łatwo podnieść go z chodnika i lekki, ale jego masa musi ułatwiać celne rzucanie. Wrzuca się kamyk do pola numer 1 i wskakuje do tego pola, zawraca, podnosi się kamień i wskakuje z powrotem na start — na wprost pierwszego prostokąta. Podczas skakania (koniecznie na jednej nodze!) nie można nadepnąć na żadną linię. Jeżeli udało się wykonać pierwszą kolejkę, wrzuca się kamyk do prostokąta numer 2. Znowu skacze się od pola do pola, aż do prostokąta, w którym leży kamyk. Należy go podnieść, obrócić się i skacze się (wciąż na jednej nodze) z powrotem. Po przekroczeniu pola numer 5 można skakać na dwóch nogach (jedną nogą w polu czwartym, drugą w piątym). To chwila odpoczynku dla nóg.

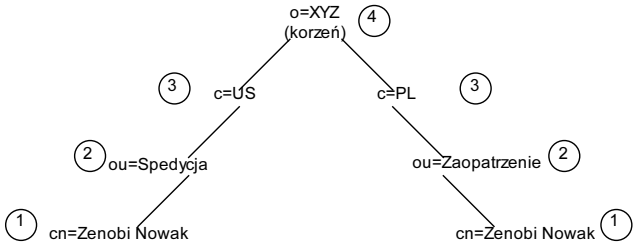
Warto przeanalizować drogę, wzdłuż której się skacze podczas gry w klasy. Wykonuje się wtedy uporządkowany zestaw ruchów, zależny od reguł gry. Wkracza się w pola

i opuszcza się je w ustalonej z góry kolejności, podnosi się kamyk (informację) i powraca do punktu wyjścia. Powyższy przykład ma pomóc w zrozumieniu sposobu, w jaki zorganizowano katalog Oracle Internet Directory i jak rozmieszczone są w nim informacje.

Każdy zbiór informacji wewnątrz katalogu jest nazywany *wpisem*, a każdy z wpisów jest identyfikowany nazwą wyróżniającą (ang. *Distinguished Name*). Podobnie jak numer identyfikował i określał położenie pola w grze w klasy, tak nazwa wyróżniająca identyfikuje wpis i definiuje lokalizację informacji reprezentowanych przez ten wpis. Zbiór wpisów i ich nazw wyróżniających jest przechowywany w strukturze hierarchicznej katalogu, którą nazywa się drzewem informacyjnym katalogu (*Directory Information Tree*, DIT).

Powrócę jeszcze na moment do struktury wcześniej opisywanego przedsiębiorstwa XYZ. Warto sprawdzić, jak wyglądałoby drzewo DIT dla dwóch różnych pracowników o tym samym imieniu i nazwisku — Zenobi Nowak — zatrudnionych w dwóch różnych wydziałach. Na rysunku 5.4 przedstawiono drzewo DIT z lokalizacją obydwóch pracowników.

Rysunek 5.4. *Hierarchia Directory Information Tree (obejmująca dwóch pracowników)*



Gdzie: o – organizacja, c – kraj, ou – jednostka organizacyjna, cn – nazwa

Ciekawa jestem, czy na podstawie poniższych informacji i rysunku 5.4 Czytelnik potrafi ustalić poprawne nazwy DN dla tych dwóch pracowników. Przypomnę jeszcze, że `o` oznacza organizację, `c` oznacza kraj, `ou` jest jednostką organizacyjna, a `cn` to nazwa ogólna.

Poniżej poziomu `cn` umieszczono wartości atrybutów, takich jak adres poczty elektronicznej, adres biura itd. Warto jednak zapamiętać, że każdy z poziomów może przechowywać atrybuty.

Podczas składania nazwy DN zaczyna się od najniższego poziomu i przechodzi się przez kolejne węzły drzewa, aż do korzenia katalogu. Na rysunku poszczególne poziomy oznaczono numerami, wskazującymi kolejność składania nazwy wyróżniającej pracowników. Poczekam, aż Czytelnik zapisze swoje nazwy na kartce.

Oto moja odpowiedź: `cn=Zenobi Nowak`, `ou=spedycja`, `c=PL`, `o=XYZ` oraz `cn=Zenobi Nowak`, `ou=zaopatrzenie`, `c=Francja`, `o=XYZ`. Najniższa gałąź drzewa DIT, znana jako względna nazwa wyróżniająca (RDN), jest pierwszą częścią nazwy i znajduje się na skrajnie lewej pozycji ciągu. Dalej przechodzi się w górę drzewa (podobnie jak w grze w klasy). Element każdego kolejnego poziomu jest nazwą RDN. Tak więc następną po `Zenobi Nowak`

nazwą RDN jest wartość atrybutu `ou`. Można więc powiedzieć, że nazwa wyróżniająca jest sekwencją względnych nazw wyróżniających, rozdzielonych przecinkami. Aby zlokalizować właściwy wpis katalogu Oracle Internet Directory, należy podać kompletną nazwę DN — pojedyncza nazwa RDN nie wystarczy.

Na rysunku 5.4 przedstawiono przykład dwóch pracowników z tym samym nazwiskiem, identyfikowanych jednoznacznie dzięki temu, że są zatrudnieni w dwóch różnych wydziałach i różnych krajach. Jednak może się zdarzyć, że dwóch takich pracowników pracuje w tej samej jednostce organizacyjnej. W takim przypadku należy znaleźć inny sposób rozróżnienia pracowników. Można każdemu pracownikowi przypisać jednoznaczny numer identyfikacyjny lub uwzględniać w nazwie ogólnej wpis inicjału drugiego imienia lub też całe drugie imię.

Atrybuty

Omawiając ogólne modele katalogów LDAP wspomniałam, że każdy wpis składa się z atrybutów, a każdy atrybut zawiera typ atrybutu i jedną lub kilka wartości atrybutu. Typ atrybutu mówi o rodzaju przechowywanej informacji. Przykładowymi atrybutami wpisu pracownika są `jobTitle` (etat), `salaryAmount` (płaca), `departmentNumber` (numer działu), `telephoneNumber` (numer telefonu) i tak dalej. Wartościami atrybutu `jobTitle` mogą być ciągi `Manager` (kierownik), `Clerk` (urzędnik) czy `Database Administrator` (administrator bazy danych).

W katalogu Oracle Internet Directory można przechowywać dwa rodzaje informacji: dane aplikacji i dane operacyjne. Wartości wymienione dla atrybutu `jobTitle` można porównać do danych aplikacji. Są to informacje, które klient pobiera ze struktury katalogu. Dane operacyjne dotyczą działań wykonywanych przez serwer katalogu. Przykładowo, wartość znacznika czasowego danego wpisu wpływa na operacje serwera, ponieważ znacznik czasowy jest uwzględniany przy operacji odświeżania wszystkich serwerów usług katalogowych uruchomionych w systemie.

Atrybuty mogą posiadać jedną lub kilka wartości. Atrybut `telephoneNumber` może przechowywać więcej niż jedną wartość dla jednego pracownika, uwzględniając numer telefonu domowego, biurowego i komórkowego, podczas gdy atrybut `gender` (płeć) może posiadać tylko i wyłącznie jedną wartość. Jeżeli założono grupę dystrybucyjną poczty elektronicznej przechowującą adresy e-mail członków drużyny sportowej, odpowiedni do niej atrybut będzie atrybutem wielowartościowym.

Standard określa zbiór podstawowych atrybutów protokołu LDAP — jest on w pełni implementowany przez katalog Oracle Internet Directory. Niektóre częściej stosowane atrybuty LDAP wymieniono w tabeli 5.1.

Katalog Oracle Internet Directory udostępnia również kilka własnych atrybutów, których opis znajduje się w dodatku F podręcznika „Oracle Internet Directory Administrator/s Guide, Release 2.0.6”.

Podobnie jak atrybuty serwera usług katalogowych LDAP określają typ i składają się z wartości, tak atrybuty katalogu Oracle Internet Directory zawierają informacje o typie

Tabela 5.1. *Popularne atrybuty LDAP, używane w Oracle Internet Directory*

Atrybut	Ciąg	Definicja
commonName	cn	Ogólna nazwa wpisu.
		Przykład: cn=Jan Nowak
domainComponent	dc	Komponent w systemie DNS (Domain Name System).
		Przykład: dc=uk, dc=com, dc=org
jpegPhoto	jpegPhoto	Fotografia w formacie JPEG. Ścieżka dostępu i nazwa pliku w formacie JPEG, który ma zostać dołączony jako atrybut wpisu.
		Przykład: jpegPhoto=/photo/janusz.jpg
organization	o	Nazwa organizacji.
		Przykład: o=XYZ
organizationalUnitName	ou	Nazwa jednostki wewnętrznej organizacji.
		Przykład: ou=Kadry
owner	owner	Nazwa wyróżniająca wpisu osoby, która posiada prawa własności do danego wpisu.
		Przykład (wiersz z pliku LDIF): owner:cn=Jan Nowak,ou=Kadry,o=XYZ,c=PL
surname	sn	Nazwisko osoby.
		Przykład: sn=Nowak
telephoneNumber	telephoneNumber	Numer telefonu.
		Przykład: telephoneNumber=(+48)222-134-32 lub telephoneNumber=(+48)22213432

i wymagania składniowe, opisujące charakter przechowywanych wartości. Atrybut telephoneNumber mógłby mieć typ dopuszczający przechowywanie wartości, które składają się wyłącznie z cyfr i kresek. Inny typ atrybutu mógłby zastrzegać stosowanie wyłącznie znaków alfanumerycznych lub np. blokować drukowanie wartości atrybutu. Katalog Oracle Internet Directory obsługuje i implementuje wszystkie standardowe typy i wymagania składniowe LDAP, umożliwia też dodawanie do katalogu własnych typów.

Wcześniejszy opis modelu funkcjonalnego LDAP zawierał wzmiankę o tym, że do definiowania sposobu przedstawiania wartości atrybutu w czasie wykonywania operacji przeszukiwania stosuje się *reguły*. Przykładowo, wartości 703-555-1212 i 7035551212 mogą być interpretowane jako identyczne, w zależności od zastosowanych reguł porównywania. Katalog Oracle Internet Directory rozpoznaje i uwzględnia wszystkie standardowe reguły dopasowania LDAP:

- DistinguishedNameMatch;
- caseExactMatch;
- caseIgnorematch;
- numericStringMatch;

- IntegerMatch;
- telephoneNumberMatch.

Klasy

Dotychczas omówiłam pojęcia wpisów i atrybutów. Teraz przedstawię pojęcie klas. Na etapie definiowania wpisu przypisuje się mu jedną lub kilka klas. Klasa zawiera atrybuty i definiuje kategorię obiektów. Może posiadać ona zarówno atrybuty obowiązkowe, jak i opcjonalne.

W przypadku klasy organizationalPerson atrybutami obowiązkowymi są commonName (cn) i surname (sn), pozostałe zaś, jak telephoneNumber czy streetAddress są dostępne, lecz nie wymagane.

Instalacja katalogu Oracle Internet Directory obejmuje standardowe klasy LDAP i kilka klas wprowadzonych przez korporację Oracle. Do predefiniowanej klasy nie można dodawać atrybutów obowiązkowych, ale jest możliwe dodawanie atrybutów opcjonalnych do istniejącej klasy, definiowanie nowej klasy lub definiowanie podklasy. Czytelnik zapewne zechce się dowiedzieć, czym jest podklasa.

Hierarchia klas

Moja kolega Vinnie jest kapitanem zakładowej drużyny baseballowej. W drużynie jest 25 mężczyzn. Przed meczem Vinnie ustala skład początkowy drużyny. Wybiera 9 graczy rozpoczynających grę, wyznacza ich pozycje oraz kolejność na stanowisku pałkarza. W terminologii katalogu Oracle Internet Directory drużyna, jako całość 25 graczy, jest nadklasą dla utworzonej klasy drużynaBaseballa. Grupa dziewięciu graczy jest podklasą wyprowadzoną z klasy drużynaBaseballa. Podklasa dziedziczy wszystkie atrybuty klasy, z której została wyprowadzona. Wpisy w katalogu Oracle Internet Directory mogą dziedziczyć atrybuty z wielu klas.

Wyróżnia się jedną, specyficzną klasę, zwaną klasą szczytową (ang. *top*). Klasa ta nie posiada nadklas i jest jedną z klas bazowych dla wszystkich klas strukturalnych. Każdy wpis w katalogu dziedziczy atrybuty klasy szczytowej. W katalogu Oracle Internet Directory klasa szczytowa ma jedną, obowiązkową klasę o nazwie objectClass. Klasa ta posiada również kilka atrybutów opcjonalnych, które wymieniono w tabeli 5.2.

Tabela 5.2. *Opcjonalne atrybuty klasy szczytowej*

Atrybut	Opis
orclGuid	Identyfikator globalny wpisu, niezmienny w czasie istnienia wpisu
creatorsName	Nazwa twórcy obiektu klasy
createTimestamp	Czas utworzenia obiektu klasy
orclACI	Modyfikowalny przez użytkownika, opcjonalny atrybut reprezentujący informacje listy kontroli dostępu
orclEntryLevelACI	Zawiera dyrektywy listy kontroli dostępu. Atrybut wielowartościowy

Dostępne są trzy rodzaje klas: klasa abstrakcyjna, strukturalna i pomocnicza. Klasy abstrakcyjne są traktowane jak klasy wirtualne i nie mogą być jedynymi klasami przypisanymi do wpisu. Klasa szczytowa jest klasą abstrakcyjną i jest nadklasą dla wszystkich pozostałych klas katalogu Oracle Internet Directory.

Większość klas tworzących katalog Oracle Internet Directory są klasami o charakterze strukturalnym. Klasy strukturalne definiują rodzaje klas, jakie mogą zostać utworzone pod daną klasą. Przykładowo, reguła struktury drzewa Directory Information Tree może określać, że wszystkie obiekty zlokalizowane poniżej klasy `osoba` muszą zawierać fizyczne cechy klasy `osoba`. Tak więc, nie byłoby możliwe umieszczenie klasy `adres` bezpośrednio pod klasą `osoba`. Dozwolone byłoby jednak umieszczenie pod nią klas `kolorWłosów`, `kolorOczu`, `płeć` i innych. Warto wiedzieć, że katalog Oracle Internet Directory w aktualnej wersji nie wymusza stosowania reguł dotyczących struktury. Trudno ocenić stosowność i prawidłowość reguł, jeżeli nie są one przestrzegane. Być może w niedalekiej przyszłości funkcja ta zostanie zaimplementowana.

Klasy pomocnicze służą do rozszerzania istniejących list atrybutów wpisu, kiedy jest niepożądane ponowne definiowanie istniejącej klasy. Istnieje możliwość zaistnienia sytuacji, w której po zdefiniowaniu wpisu należącego do dwóch klas trzeba dodać do wpisu atrybuty, które nie istnieją w żadnej z tych klas. W takiej sytuacji można utworzyć klasę pomocniczą i umieścić w niej nowe klasy. Następnie owe klasy można powiązać z wpisem, nie wpływając na dotychczas przypisane klasy. Przykładowo, istnieje klasa o nazwie `konie`, obejmująca konie arabskie oraz tarpany, koniki polskie itp. W pewnej chwili potrzebna okazuje się klasa obejmująca wyłącznie konie wyścigowe. Modyfikacja istniejącej klasy nie jest dobrym rozwiązaniem. Dlatego tworzymy nową klasę pomocniczą o nazwie `konieWyścigowe`.

Wpisy podrzędne i schematy

W bazie danych Oracle, w obszarze określanym mianem słownika bazy danych są przechowywane metadane. Są to informacje o budowie każdego z obiektów bazy danych, łącznie z jego nazwą, rozmiarem, typem i innymi niezbędnymi danymi. Katalog Oracle Internet Directory przechowuje takie metadane, jak definicje klas, atrybutów, składni i reguł dopasowania. Jak wspomniałam wcześniej, są one przechowywane w strukturze zwanej schematem słownika. Schemat słownika przechowuje informacje w specjalnej klasie wpisu, który jest nazywany podwpisem (wpisem podrzędnym). Wpis podrzędny jest określany jako `subSchemaSubentry`, co zdefiniowano w wersji 3. standardu LDAP. Dozwolone są modyfikacje wpisu `subSchemaSubentry`, polegające na dodawaniu nowych klas i obiektów. Katalog Oracle Internet Directory obsługuje określony z góry zestaw reguł dopasowania i składni. Nie jest możliwe dodawanie nowych, własnych składni ani reguł dopasowania.

Rozpraszanie katalogu

W czasie, gdy korporacja Oracle rozpoczynała swoją działalność, bazy danych zasadniczo rezydowały na pojedynczym, centralnym komputerze. W miarę rozwoju technologii baz danych i oprogramowania oraz sprzętu komputerowego uzyskano możliwość rozpraszania bazy danych pomiędzy dyskami, a teraz można nawet rozpraszać bazę danych pomiędzy komputerami, których fizyczna lokalizacja jest prawie dowolna.

Część bazy danych może być przechowywana w Rzymie we Włoszech, podczas gdy inne części mogą być rozmieszczone na komputerach w Wenecji. Z punktu widzenia użytkowników bazy danych jest ona logicznie jedną, nierozerwalną strukturą.

W podobny sposób można fizycznie rozproszyć dane przechowywane w katalogu Oracle Internet Directory pomiędzy kilka serwerów, a mimo to będą one stanowiły jedną, logiczną całość. Rozproszenie danych katalogu pomiędzy kilka serwerów daje dwie zasadnicze korzyści. W ten sposób zwiększa się maksymalny, możliwy do przechowania rozmiar katalogu i równocześnie zmniejsza nakład pracy związany z obsługą klientów, przypadający na każdy z serwerów. Dodatkową korzyścią jest usuwanie potencjalnych „wąskich gardeł" obsługi klientów. Ważne też jest, że część użytkowników może kontynuować pracę w przypadku niedostępności jednego z serwerów.

Aby dokonać rozproszenia katalogu, należy podzielić dane przechowywane w katalogu na jednostki zwane kontekstami nazw. Każdy kontekst nazw w rzeczywistości jest poddrzewem, umieszczonym na jednym serwerze i posiadającym wpis odgrywający rolę korzenia. Poddrzewo rozszerza się w dół do węzłów końcowych lub węzłów, które są referencjami do podrzędnych kontekstów nazw. Rozmiar kontekstu nazw nie jest ograniczony z góry, może więc mieć wielkość pojedynczego wpisu lub zawierać całe drzewo informacyjne DIT katalogu.

Aby zdecentralizować katalog, można skorzystać z procesu replikacji lub partycjonowania katalogu (ang. *partitioning*). Replikacja obiektu oznacza wykonanie jego dokładnej kopii, podczas gdy partycjonowanie obiektu oznacza podział obiektu na rozłączne części. Można więc dokonać rozproszenia katalogu przez kopiowanie zawartości kontekstu nazw na inne komputery lub przez podział kontekstu na jeden lub kilka rozłącznych kontekstów nazw i umieszczenie każdego z nich na osobnym komputerze.

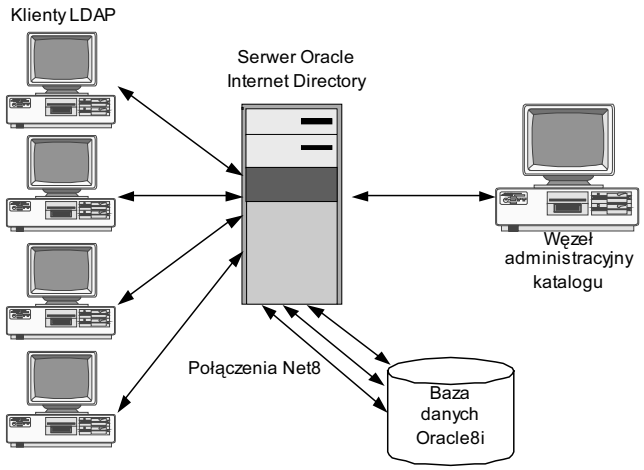
Oracle Internet Directory i Net8

Można się zastanawiać, co mają wspólnego ze sobą moduły Oracle Listener, Oracle Internet Directory i Oracle Names Server. Można odpowiedzieć, że wszystkie one są aplikacjami, które można uruchomić na serwerze Oracle8i — i byłaby to odpowiedź całkiem poprawna. Najpełniejszym określeniem jest jednak, że wszystkie te elementy komunikują się za pośrednictwem Net8. Katalog Oracle Internet Directory może — ale nie musi — działać na tym samym komputerze, co baza danych Oracle, ale w każdym przypadku komunikacja pomiędzy katalogiem i bazą danych odbywa się za pośrednictwem Net8, podobnie jak komunikacja pomiędzy bazą danych, a procesem nasłuchującym, czy serwerem Oracle Names. Do komunikacji z bazami danych Oracle8i służą Net8 oraz interfejs OCI.

Komponenty

Na rysunku 5.5 przedstawiono związek pomiędzy klientami LDAP i katalogiem Oracle Internet Directory oraz częścią platformy Net8, która umożliwia katalogowi komunikowanie się z bazą danych Oracle8i. Klienty LDAP zgłaszają żądania serwerowi

usług katalogowych Oracle Internet Directory. Serwer nawiązuje z kolei połączenie z katalogiem Oracle8i i realizuje wyszukiwanie żądanej informacji. Po zakończeniu wyszukiwania jego wynik jest zwracany do klienta LDAP za pośrednictwem serwera Oracle Internet Directory.



Rysunek 5.5. *Oracle Internet Directory i komunikacja za pośrednictwem Net8*

Omówię teraz właściwe komponenty zaangażowane w ten proces. W tabeli 5.3 zaprezentowano komponenty katalogu Oracle Internet Directory wraz z ich opisem.

Standardowo serwer katalogu Oracle Internet Directory jest instalowany wraz z wpisem konfiguracyjnym. Każdy wpis konfiguracyjny nosi nazwę `configset`. Wpis `configset` przechowuje parametry konfiguracyjne konkretnego egzemplarza serwera usług katalogowych. Polecenia uruchomienia lub zatrzymania serwera z poziomu narzędzia Oracle Internet Directory Control (`oidctl`) zawierają referencję do jednego z wpisów `configset`. Monitor katalogu Oracle Internet Directory realizuje żądania inicjowane poleceniami `oidctl` i wykorzystuje informacje przechowywane we wpisie konfiguracyjnym. Standardowy, domyślnie instalowany wpis konfiguracyjny nosi nazwę `configset0`. Dzięki dostępności tego standardowego wpisu konfiguracyjnego jest możliwe uruchomienie serwera Oracle Internet Directory bezpośrednio po jego instalacji. Konfiguracja i dostosowanie serwera do potrzeb środowiska oraz przeglądanie i dodawanie nowych wpisów konfiguracyjnych jest możliwe dzięki programowi OID Manager lub narzędziom wiersza poleceń. W dalszej części niniejszego rozdziału przedstawię jeszcze pewne fakty dotyczące programu OID Manager.

W razie zakończenia działania serwera w trybie awaryjnym monitor OID Monitor dokonuje ponownego jego uruchomienia. Po uruchomieniu serwera monitor umieszcza wpis w rejestrze poszczególnych serwerów katalogu i aktualizuje dane w tabelach informacji o procesach. Wpis rejestracyjny i wpisy w tabeli procesów są usuwane w momencie zatrzymania serwera przez monitor OID. Jeżeli serwer jest niespodziewanie zatrzymany, po ponownym jego uruchomieniu przez monitor OID następuje zaktualizowanie wpisu rejestracyjnego. Jest też dodawany nowy znacznik czasowy, odpowiedni dla momentu uruchomienia serwera.

Tabela 5.3. *Komponenty Oracle Internet Directory*

Komponent	Opis
Serwer LDAP	Obsługuje żądania usług katalogowych za pośrednictwem pojedynczego procesu dyspozytora katalogu Oracle Internet Directory, nasłuchującego na określonym porcie TCP/IP (domyślnie port 389 dla połączeń bez SSL i port 636 dla połączeń z SSL). Jeżeli na danym komputerze działa kilka serwerów LDAP, każdy z nich musi nasłuchiwać na innym porcie. Procesy serwera i dyspozytora katalogu Oracle Internet Directory korzystają z architektury wielowątkowej
Serwer replikacji	Śledzi i przesyła pomiędzy replikowanymi serwerami dane dotyczące zmian zawartości katalogu Oracle Internet Directory. Na danym węźle może działać tylko jeden serwer replikacji. Zastosowanie serwera replikacji jest opcjonalne
Baza danych Oracle8i	Przechowuje zawartość katalogu. Baza danych może działać na tym samym węźle, co serwery LDAP, albo na osobnych węzłach. Oracle zaleca stosowanie osobnych baz danych dla serwera Oracle8i i serwera Oracle Internet Directory
Narzędzie OID Control (`oidctl`)	Służy do uruchamiania i zatrzymywania serwerów usług katalogowych. Polecenia `oidctl` są interpretowane przez monitor OID Monitor (`oidmon`). Program `oidctl` komunikuje się z monitorem `oidmon` i umieszcza komunikaty (wraz z parametrami konfiguracyjnymi wymaganymi do uruchomienia egzemplarzy OID) w tabelach serwera OID
Monitor OID (`oidmon`)	Inicjuje, monitoruje i zatrzymuje procesy serwerów LDAP. Monitor OID steruje serwerem replikacji (o ile jest on zainstalowany)

W systemie jest przechowywany plik o nazwie *ORACLE_HOME/ldap/log/oidmon.log*, w którym są rejestrowane działania monitora OID Monitor. Mechanizmy służące do określania stanu poszczególnych egzemplarzy serwera usług katalogowych są zależne od systemu operacyjnego.

Żądania klientów

Aby prześledzić proces zgłaszania i obsługi żądania klienta LDAP, najpierw należy poznać składowe serwera usług katalogowych LDAP. Jednym z nich jest proces dyspozytora katalogu Oracle Internet Directory, który prowadzi nasłuch na porcie przeznaczonym do obsługi poleceń LDAP. Proces dyspozytora OID obsługuje zarówno funkcje procesu nasłuchującego, jak i procesu dyspozytora (procesu rozdzielającego żądania).

W momencie startu serwera usług katalogowych LDAP jest tworzony proces serwera. Liczba tworzonych procesów serwera jest określana wartością parametru konfiguracyjnego `orclserverprocs`. Podczas uruchamiania kilku procesów serwera katalog OID może wykorzystać właściwości systemów wieloprocesorowych. Aby obsłużyć żądanie klienta, dla każdej operacji jest tworzony wątek roboczy. Maksymalna liczba połączeń z bazą danych, które mogą być utworzone w razie potrzeby przez każdy z procesów serwera, jest określana wartością parametru `orclmaxcc`. Jego wartość domyślna wynosi `10`.

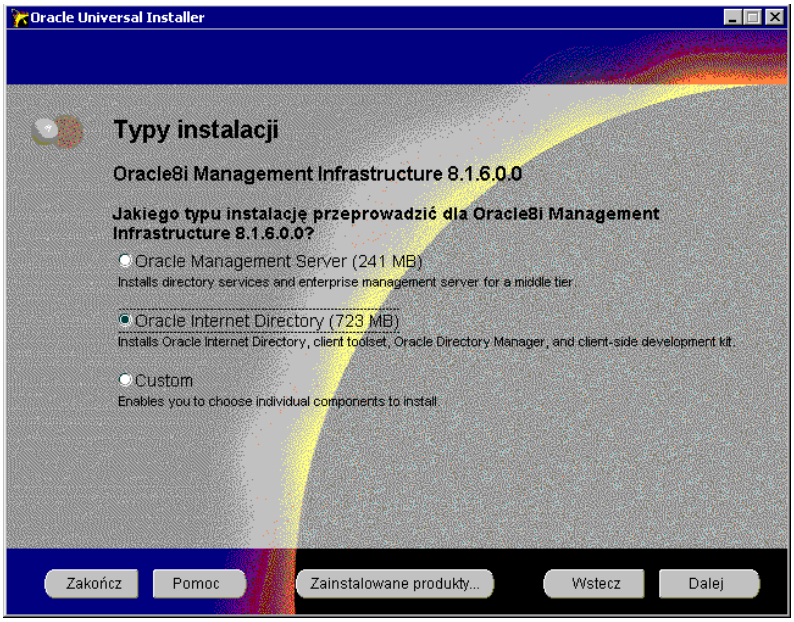
Poniżej przedstawiono sekwencję zdarzeń, która następuje po zgłoszeniu żądania wyszukiwania przez klienta.

1. Klient wprowadza żądanie wyszukiwania, uwzględniające jedną lub kilka z poniższych opcji:
   - SSL: *Protokół Secure Sockets Layer* umożliwia szyfrowanie transmisji i uwierzytelnianie stron połączenia. Jeżeli klient nie wykorzystuje SSL, żądanie klienta jest przesyłane jako tekst jawny, łatwy do przechwycenia przez „szperacze" sieciowe;
   - typ użytkownika: określenie typu użytkownika — konkretnego użytkownika lub użytkownika anonimowego — określa uprawnienia wymagane do wykonania danej operacji;
   - filtry: służą do zawężania dziedziny poszukiwań i mogą zawierać warunki i operacje logiczne, takie jak `AND`, `OR` oraz `NOT`. Można wykorzystywać operatory `GREATER THAN`, `EQUAL TO` i `LESS THAN`.
2. W celu wykonania żądania klient może wykorzystać program OID Manager lub narzędzia wiersza poleceń. Jeżeli jest stosowany program OID Manager, do zgłoszenia żądania jest wykorzystywana funkcja zapytania interfejsu Java Native Interface. Interfejs ten z kolei wywołuje funkcje interfejsu programistycznego w języku C. Narzędzia uruchamiane z wiersza poleceń wywołują bezpośrednio funkcje interfejsu w języku C.
3. Żądanie jest przesyłane do serwera usług katalogowych za pośrednictwem protokołu LDAP.
4. Serwer dokonuje — za pomocą procesu nazywanego wiązaniem — uwierzytelnienia użytkownika i sprawdzenia list kontroli dostępu (*Access Control List*, ACL) w celu weryfikacji posiadania przez użytkownika uprawnień wymaganych do zrealizowania żądanej operacji.
5. Żądanie wyszukiwania jest konwertowane z języka protokołu LDAP do postaci wywołań interfejsu Oracle Call Interface (OCI) — obsługiwanych przez serwer usług katalogowych — i przesyłane za pośrednictwem Net8 do bazy danych Oracle8i.
6. Baza danych Oracle8i analizuje żądanie, podobnie jak każde inne zapytanie kierowane do bazy danych, następnie pobiera z tabel informacje, które są przekazywane z powrotem do serwera usług katalogowych, dalej do wywołań interfejsu języka C i ostatecznie do klienta.

Instalacja katalogu Oracle Internet Directory

Wydaje mi się, że nie istnieje prosty sposób na określenie sposobu początkowej instalacji katalogu Oracle Internet Directory na podstawie zestawu dokumentacji dostarczonej przez Oracle. Chciałabym więc przeznaczyć trochę czasu na omówienie tych zagadnień i przeprowadzenie Czytelnika przez proces instalacji.

Aby przeprowadzić instalację katalogu OID, należy uruchomić instalator Oracle Universal Installer. W odpowiedzi na pytanie o wybór oprogramowania do instalacji, które pojawia się w pierwszym oknie instalatora, należy wskazać pozycję trzecią, *Oracle8i Management Infrastructure 8.1.6.0.0*. Po dokonaniu wyboru naciska się przycisk *Next*, w efekcie czego ukazuje się okno kreatora instalacji Installation Type (rysunek 5.6).



Rysunek 5.6. *Oracle Universal Installer — okno wyboru rodzaju instalacji*

Jeżeli w systemie już zainstalowano wydanie 2. bazy danych Oracle8i, podczas instalacji OID trzeba odpowiedzieć na pytanie, czy będzie wykorzystywana istniejąca baza danych, czy też instalator ma utworzyć nową bazę. W razie wykorzystania tej ostatniej możliwości w następnym oknie instalatora należy podać identyfikator SID wykorzystywanej bazy danych. Następnie można podjąć decyzję, czy wartość atrybutu `userPassword` (hasło użytkownika) ma być szyfrowana. Ja nacisnęłam przycisk *No*, aby nie szyfrować hasła, więc w następnym oknie instalatora znalazła się informacja, że nazwą wyróżniającą obszaru będzie `cn=orcladmin`. W oknie tym była widoczna również długość dotychczasowego hasła, ale jego wartość nie była wyświetlona. Jak się później Czytelnik przekona, domyślnym hasłem jest ciąg `welcome`.

W kolejnym oknie instalatora istnieje możliwość wybrania przybliżonej liczby wpisów, które mają być przechowywane w katalogu. Zakres tej wartości może wynosić od najwyżej 10000 do ponad 1000000. Wreszcie następuje wyświetlenie żądania potwierdzenia danych dotyczących liczby i rodzajów instalowanych produktów, po czym dochodzi do właściwej instalacji. Postęp procesu instalacji można śledzić obserwując komunikaty wyświetlane w oknie *Confguration Tools*. Instalacja może wydawać się zadziwiająco prostym zadaniem, gdy już wiadomo, jak ją przeprowadzić.

Warto zauważyć, że chociaż informacje wyświetlane w oknach instalatora sugerują, że instalacja wymaga 723 megabajtów przestrzeni dyskowej, to w przypadku istniejącej

instalacji produktów Oracle8i jest wymagany znacznie mniejszy obszar. Wyświetlany rozmiar wymaganej przestrzeni jest obliczony dla wykonywania pełnej instalacji Oracle8i.

Warto przeanalizować zawartość zestawu narzędzi OID i sprawdzić, co i jak można zrobić z nowo zainstalowanym katalogiem.

Narzędzia Oracle Internet Directory

Komunikowanie się i administrowanie serwerem katalogu Oracle Internet Directory jest możliwe za pomocą kilku narzędzi, obsługiwanych z wiersza poleceń oraz za pomocą jednego narzędzia wyposażonego w graficzny interfejs użytkownika. Są to:

- narzędzia wiersza poleceń, które komunikują się z użytkownikiem za pomocą plików tekstowych w formacie LDAP *Data Interchange Format* (LDIF) i służą do manipulowania wpisami i atrybutami;
- narzędzia służące do wydajnego ładowania, modyfikacji i usuwania dużej liczby wpisów oraz do kopiowania danych z bazy informacyjnej katalogu do plików LDIF, co jest przydatne w przypadkach masowego ładowania danych;
- narzędzie OID Control, służące do uruchamiania i zatrzymywania procesów serwera OID;
- program zarządzający katalogiem Catalog Management;
- narzędzie służące do ustalania i zmiany hasła OID Database Password (domyślnym hasłem katalogu Oracle Internet Directory jest *ODS*);
- narzędzie wyposażone w graficzny interfejs użytkownika — OID Manager — służące do zarządzania katalogiem.

W kolejnych podrozdziałach przedstawiłam możliwości udostępniane przez te narzędzia.

Narzędzia obsługiwane z wiersza poleceń

W rozdziale 3. opisywałam funkcje procesu nasłuchującego i jego narzędzia sterującego Listener Control (`lsnrctl`). Przypomnę tu, że w celu uruchomienia tego narzędzia należy wywołać je z poziomu systemowego interpretera poleceń — w oknie MS-DOS w przypadku systemu Windows NT lub w powłoce systemowej UNIX. W ten sam sposób uruchamia się narzędzie sterujące serwera Oracle Names, `namesctl`. W przypadku serwera Oracle Internet Directory można dysponować różnymi narzędziami obsługiwanymi z wiersza poleceń, służącymi do rozmaitych czynności zarządzających wpisami katalogu. Niektóre narzędzia umożliwiają dostęp do indywidualnych wpisów, inne operują całymi ich grupami. Jeszcze inne narzędzia umożliwiają wykonywanie operacji związanych z administracją. Najpierw opiszę narzędzia obsługiwane z wiersza poleceń, natomiast narzędzie z graficznym interfejsem użytkownika — OID Manager — przedstawię w końcowej części niniejszego rozdziału.

Narzędzia manipulacji wpisami i atrybutami

Pierwszy, omawiany tu zestaw narzędzi, służy do manipulowania wpisami i atrybutami. Aby móc je wykorzystać, tworzy się wpisy w formacie LDAP *Data Interchange Format* i umieszcza się je w pliku tekstowym. Każdy wpis w takim pliku składa się z nazwy wyróżniającej oraz jednego lub kilku atrybutów i ich wartości, a każdy atrybut powinien się znaleźć w osobnym wierszu. Tabela 5.4 zawiera nazwy i opis narzędzi operujących plikami LDIF, obsługiwanych z poziomu wiersza poleceń.

Tabela 5.4. *Narzędzia wiersza poleceń operujące plikami LDIF*

Narzędzie	Opis
`ldapsearch`	Przeszukuje katalog w celu odszukania określonych wpisów
`ldapbind`	Uwierzytelnia użytkownika-klienta wobec serwera usług katalogowych
`ldapadd`	Dodaje do katalogu jeden wpis
`ldapaddmt`	Dodaje do katalogu kilka wpisów równocześnie z zastosowaniem mechanizmu wielowątkowości
`ldapmodify`	Tworzy, aktualizuje i usuwa dane atrybutu określonego wpisu
`ldapmodifymt`	Modyfikuje kilka wpisów równocześnie z zastosowaniem mechanizmu wielowątkowości
`ldapdelete`	Usuwa wpisy
`ldapcompare`	Sprawdza wpis pod kątem określonej wartości atrybutu
`ldapmoddn`	Modyfikuje nazwę DN lub RDN wpisu, zmienia nazwę wpisu lub poddrzewa, przenosi wpis lub poddrzewo do nowej gałęzi

Narzędzia manipulujące grupami wpisów

Drugi rodzaj narzędzi wiersza poleceń obejmuje narzędzia służące do realizowania zadań, wymagających manipulacji dużą liczbą wpisów katalogu. W ten sposób można wykonywać masowe aktualizacje i modyfikacje wpisów, a także ładowanie dużej liczby wpisów oraz eksport dużej liczby wpisów do pliku w formacie LDIF, który służy następnie jako źródło masowego ładowania danych dla innego katalogu. Narzędzia tej grupy wraz z opisem ich przeznaczenia są wymienione w tabeli 5.5.

Tabela 5.5. *Narzędzia operujące na dużej ilości danych*

Narzędzie	Opis
`bulkload`	Ładuje do katalogu Oracle Internet Directory dużą liczbę wpisów zdefiniowanych w plikach LDIF
`ldifwrite`	Kopiuje dane z bazy informacyjnej katalogu do pliku LDIF. Każdy serwer usług katalogowych, zgodny z protokołem LDAP, może następnie taki plik odczytać. Narzędzie to w połączeniu z narzędziem `bulkload` umożliwia masowe ładowanie danych. Narzędzie `ldifwrite` może być służyć do archiwizacji zawartości katalogu lub jego części
`bulkmodify`	Modyfikuje równocześnie dużą liczbę wpisów katalogu
`bulkdelete`	Służy do szybkiego usuwania poddrzewa

Narzędzie Oracle Internet Directory Control

Narzędzie OID Control (`oidctl`) służy do uruchamiania i zatrzymywania procesu serwera Oracle Internet Directory. Polecenia wydawane w programie `oidctl` są interpretowane i wykonywane przez proces monitora OID Monitor. Oto składnia przykładowego polecenia, uruchamiającego serwer LDAP:

```
oidctl connect=<NAZWA_USŁUGI_SIECIOWEJ> server=OIDLDAPD
instance=<NUMER_INSTANCJI_SERWERA> start
```

Należy zapamiętać, że nazwa serwera jest wartością stałą, równą `OIDLLDAPD` (jeżeli nie korzysta się z serwera replikacji) lub `OIDREPLD` (jeżeli korzysta się z serwera replikacji). Numer instancji jest arbitralnie określanym numerem, przypisanym do egzemplarza serwera. Polecenie `oidctl` służy do aktywacji narzędzia OID Control. Parametrami tego polecenia, które należy określić, są nazwa usługi i numer instancji. Pozostałe parametry są opcjonalne. Przykładowo, katalog Oracle Internet Directory o nazwie usługi `OID1` i numerze instancji równym `2` jest uruchamiany następującym poleceniem:

```
oidctl connect=OID1 server=OIDLDAPD instance=2 start
```

Próba uruchomienia procesu serwera Oracle Internet Directory może zakończyć się komunikatem o błędzie „NLS_LANG not set in environment. Please set NLS_LANG to the appropriate UTF8 character set" (Nie ustawiono zmiennej środowiskowej NLS_LANG. Ustaw zmienną NLS_LANG tak, aby wskazywała odpowiedni zestaw znaków). W takim przypadku należy z poziomu powłoki systemowej wydać polecenie `set NLS_LANG=POLISH_POLAND.UTF8`, a następnie ponownie wykonać polecenie uruchamiające serwer. Zdarzało się bowiem, że baza danych musiała zostać utworzona z odpowiednim zestawem znaków UTF8 przed pomyślnym uruchomieniem procesu, w innych przypadkach wystarczało ustawienie zmiennej `NLS_LANG`. Do czasu, w którym pisałam niniejszą książkę, problem ten nie został rozwiązany.

Warto zapoznać się z poszczególnymi parametrami narzędzia `oidctl`.

Wszystkie dostępne parametry i ich opis zostały wymienione w tabeli 5.6.

Narzędzie Catalog Management

Aby udostępnić atrybuty wpisów podczas przeprowadzania operacji wyszukiwania, katalog Oracle Internet Directory wykorzystuje indeksy. Standardowo, aby wyliczyć dostępne w wyszukiwaniu atrybuty po zainstalowaniu katalogu OID, jest definiowany wpis `cn=CATALOGS`. Zindeksowany może być każdy atrybut oznaczony regułą dopasowania tożsamościowego (ang. *equal*). Aby udostępnić dodatkowe atrybuty podczas przeprowadzania wyszukiwania, należy za pomocą narzędzia Catalog Management dodać do katalogu odpowiednie wpisy.

Aby skorzystać z narzędzia Catalog Management w systemie UNIX, należy najpierw wyzerować zmienną środowiskową `LANG`. W tym celu w powłoce Korn shell należy wykonać polecenie `unset lang`. W powłoce C shell trzeba zaś wpisać `unsetenv lang`.

Tabela 5.6. *Parametry narzędzia Oracle Internet Directory Control (oidctl)*

Parametr	Opis
net_service_name	Nazwa określona w pliku *tnsnames.ora* (o ile taki istnieje). Plik ten umieszczony jest w katalogu *ORACLE_HOME/network/admin*
server	Typ uruchamianego serwera. Dopuszczalnymi wartościami są OIDLDAPD i IODREPLD. Wielkość znaków w ciągach określających tę wartość nie jest istotna
server_instance_number	Numer instancji uruchamianego serwera. Dopuszczalnymi wartościami są liczby, mieszczące się w zakresie od 0 do 1000
configset_number	Identyfikator wpisu konfiguracyjnego configset dla uruchamianego serwera. Domyślną wartością tego parametru jest configset0. Dopuszczalne są wartości od configset0 do configset1000.
-p numer_portu	Określa numer portu instancji serwera. Domyślnie numerem tym jest 389
-debug poziom_diagnostyczny	Określa poziom diagnostyczny uruchamianej instancji serwera LDAP
-h nazwa_komputera	Określa nazwę komputera, na którym jest uruchomiana instancja serwera
-l	Włącza lub wyłącza rejestrowanie w pliku dziennika zmian podlegających replikacji. W przypadku braku tego parametru rejestrowanie jest włączone. Dopuszczalnymi wartościami są TRUE i FALSE, przy czym wartością domyślną jest TRUE
-server n	Określa liczbę procesów serwera nasłuchujących na danym porcie
start	Uruchamia serwer identyfikowany pozostałymi parametrami
stop	Zatrzymuje serwer identyfikowany pozostałymi parametrami

Po zakończeniu sesji programu Catalog Management można, oczywiście, przywrócić poprzednią wartość zmiennej LANG. Za pomocą narzędzia Catalog Management można dodać lub usunąć indeks do danego atrybutu. Składnia odpowiedniego polecenia jest następująca (w moim komputerze plik *catalog.sh* znajduje się w katalogu *ORACLE_HOME\ldap\bin*):

```
catalog.sh -connect <NAZWA_USŁUGI_SIECIOWEJ> <add|delete> -attr <NAZWA_ATRYBUTU> -
file <NAZWA_PLIKU>
```

Indeksowane atrybuty można określać pojedynczo. Można też podać nazwę pliku, w którym są wymienione atrybuty podlegające indeksowaniu. Atrybuty w pliku muszą być wymienione jeden pod drugim, w jednym wierszu może się znajdować jeden atrybut. W podobny sposób można usuwać indeksy dla poszczególnych atrybutów, określonych w wierszu polecenia lub pliku.

Aby uruchomić narzędzie Catalog Management, trzeba znać hasło użytkownika katalogu OID. W razie podania niewłaściwego hasła uruchomienie narzędzia jest niemożliwe. Po zakończeniu operacji indeksowania można przywrócić wartość zmiennej środowiskowej LANG, wykonując polecenia set lang=<identyfikator_języka>, a następnie export lang (w powłoce Korn) lub setenv lang <identyfikator_języka> (w powłoce C).

Narzędzie OID Database Password

Serwer OID korzysta z hasła użytkownika podczas nawiązywania połączenia z bazą danych Oracle. Jak już nadmieniałam, domyślnym hasłem jest `ODS`. Za pomocą narzędzia OID Database Password można to hasło zmienić. W przypadku systemu UNIX w tym celu należy wykonać polecenie

```
oidpasswd
```

Po uruchomieniu narzędzia OID Database Password trzeba podać aktualne hasło, propozycję nowego hasła i potwierdzenie nowego hasła. Narzędzie OID Database Password w sposób domyślny zmienia hasło tej bazy danych, która jest określana wartościami zmiennych `ORACLE_HOME` i `ORACLE_SID`. Jeżeli zmiana hasła ma dotyczyć innej bazy danych, należy skorzystać z parametru `connect=<NAZWA_USŁUGI_SIECIOWEJ>`.

Narzędzie OID Manager

Jedynym programem z zestawu narzędziowego Oracle Internet Directory, które wyposażono w graficzny interfejs użytkownika, jest OID Manager. Przed uruchomieniem tego narzędzia należy uruchomić instancję serwera usług katalogowych. Oznacza to, że przed uruchomieniem narzędzia OID Manager należy uruchomić proces demona OID Monitor. Oczywiście, obowiązuje założenie, że utworzono katalog Oracle Internet Directory, a także że w systemie zainstalowano oprogramowanie OID. W końcowej części rozdziału 9. znajdują się wskazówki dotyczące przeprowadzenia czynności, niezbędnych do utworzenia katalogu Oracle Internet Directory za pomocą narzędzia Net8 Configuration Assistant.

Uruchamianie i zatrzymywanie monitora OID

Aby uruchomić proces monitora OID, należy najpierw dokonać odpowiednich ustawień językowych. Językiem ustawianym w trakcie instalacji jest *AMERICAN_AMERICA*. Po ustawieniu odpowiedniego języka (jeżeli jest wymagany inny niż standardowy) można z poziomu systemowego interpretera poleceń wykonać polecenie `start`:

```
oidmon <connect=NAZWA_USŁUGI_SIECIOWEJ> <sleep=LICZBA_SEKUND> start
```

Parametry `connect` i `sleep` są opcjonalne. `NAZWA_USŁUGI_SIECIOWEJ` jest nazwą określoną dla serwera OID w pliku *tnsnames.ora*. Wartość parametru `sleep` określa czas wyrażony w sekundach, jaki monitor OID odczeka przed sprawdzeniem, czy z narzędzia OID Control nie napłynęły nowe żądania i przed wznowieniem zatrzymanych procesów serwera. Domyślną wartością parametru Parametr `sleep` jest 10 sekund. Aby zatrzymać proces monitora OID, należy wykonać polecenie o zaprezentowanej powyżej składni, jednakże ostatni parametr (`start`) należy zastąpić parametrem `stop`.

Po uruchomieniu monitora OID można uruchomić instancję serwera. W tym celu należy zastosować narzędzie Oracle Internet Directory Control — opisywałam to przed chwilą. Jeżeli działają już zarówno proces monitora, jak i instancja serwera OID, można uruchomić narzędzie OID Manager.

Nareszcie!

Oracle Internet Directory Manager

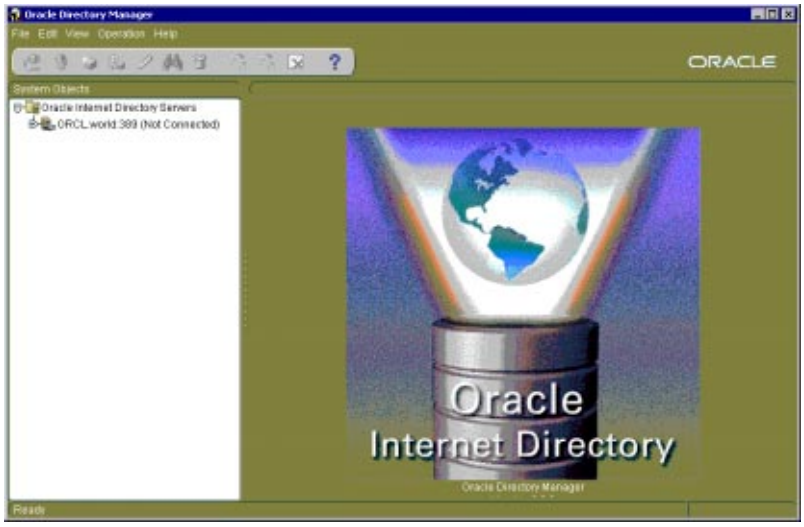
Uruchomienie programu Oracle Internet Directory Manager w systemie Windows NT polega na wybraniu z menu *Start* kolejnych poleceń: *Programy/Oracle <ORACLE_HOME>/Oracle Internet Directory/Oracle Directory Manager*. W przypadku komputera działającego pod kontrolą systemu UNIX należy natomiast wykonać polecenie `oidadmin`. Podczas pierwszej próby uruchomienia następuje wyświetlenie informacji, że najpierw trzeba nawiązać połączenie z serwerem. Należy nacisnąć przycisk *OK*.

Następnie jest wyświetlane żądanie podania identyfikatora oraz hasła użytkownika. Oto lista identyfikatorów i haseł obowiązujących podczas pierwszego logowania:

- aby zalogować się jako administrator w celu skonfigurowania właściwości związanych z SSL, należy w polu *User* wpisać `cn=orcladmin`;
- aby dokonać logowania anonimowego należy pozostawić pole *User* puste;
- jeżeli wcześniej, za pomocą narzędzi tekstowych, ustawiono odpowiednie wpisy użytkowników, można określić odpowiednią nazwę użytkownika naciskając przycisk *Browse* i wybierając odpowiednią pozycję, albo też wpisać nazwę wyróżniającą wpisu danego użytkownika w formie `cn=<ID_UŻYTKOWNIKA>,ou=<JEDNOSTKA_ORGANIZACYJNA>,o=<ORGANIZACJA>,c=<KRAJ>`.

Jeżeli podczas instalacji ustalono hasło superużytkownika, można je wykorzystać do nawiązania połączenia w roli administratora. Domyślnym hasłem narzędzia OID Manager jest hasło `welcome`. Logowanie anonimowe nie wymaga wypełniania pola *Password*.

Po pomyślnym zalogowaniu się jako administrator można zobaczyć okno powitalne (rysunek 5.7).



Rysunek 5.7. *Okno powitalne programu Oracle Directory Manager*

Za pomocą programu Oracle Directory Manager można definiować klasy, atrybuty, punkty kontroli dostępu oraz wpisy. Możliwa jest również konfiguracja zarządzania

wpisu, odświeżanie wpisów i wpisów podrzędnych, a także usuwanie indeksów atrybutów. Można tworzyć pliki pierwotne i pliki nowe, będące dokładnymi kopiami plików istniejących. Operacje te są dostępne za pomocą rozwijanych menu poleceń lub przycisków paska narzędziowego.

Po zakończeniu konfiguracji katalogu Oracle Internet Directory są dostępne pozycje *Subtree Access Management*, *Entry Management*, *Schema Management* i *Audit Log Management*. Dostępne są opcje konfiguracyjne SSL i opcje dostępu niezabezpieczonego.

W tabeli 5.7 wyliczono i krótko opisano ikony paska narzędziowego, zaczynając od ikony położonej z lewej strony. Ikona ta wyobraża wtyczkę.

Tabela 5.7. *Ikony programu Oracle Directory Manager*

Ikona	Opis
Wtyczka	Nawiązanie połączenia (lub rozłączenie) z serwerem wybranym za pomocą panelu nawigacyjnego
Zakręcona strzałka	Odświeżenie (aktualizacja) danych obiektów, które nie są wpisami, a które są przechowywane w pamięci podręcznej — uwzględnienie aktualizacji zawartości bazy danych
Twórz	Dodanie nowego obiektu
Twórz na podstawie	Dodanie nowego obiektu na podstawie wskazanego obiektu (szablonu)
Ołówek	Edycja lub modyfikacja właściwości obiektu
Lornetka	Wyszukiwanie obiektu
Kosz	Usuwanie obiektu
Odśwież wpis	Odświeżenie (aktualizacja) danych wpisu przechowywanych w pamięci podręcznej — uwzględnienie aktualizacji zawartości bazy danych
Odśwież wpis podrzędny	Odświeżenie (aktualizacja) danych wpisów podrzędnych przechowywanych w pamięci podręcznej — uwzględnienie aktualizacji zawartości bazy danych
Usuń indeks	Usunięcie indeksu atrybutu. Wymaga potwierdzenia wykonania operacji
Pomoc	Wyświetla okno pomocy

Informacje prezentowane w bieżącym rozdziale pochodziły z poniższej książki:

> Timothy A. Howes i Mark C. Smith , „LDAP Programming Directory-Enabled Applications with Lightweight Directory Access Protocol”, Macmilian Technology Series, 1997.