



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



UPGOW – Uniwersytet Partnerem Gospodarki Opartej na Wiedzy
Uniwersytet Śląski w Katowicach, ul. Bankowa 12, 40-007 Katowice, <http://www.us.edu.pl>

Zadanie 43

Bazy danych w środowisku Oracle

Instalacja narzędzi Oracle: konfiguracja serwera i klienta

Opracowanie: Andrzej Grzybowski

Spis dodatków

| | | |
|----|---|----|
| A. | Instalacja SZBD Oracle 10g w systemie operacyjnym Windows XP Professional z wybraną opcją tworzenia startowej bazy danych | 3 |
| B. | Obsługa podstawowych usług serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional | 46 |
| C. | Instalacja i konfiguracja klienta serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional | 53 |
| D. | Konfiguracja klienta serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional dla połączeń skryptów PHP z bazą danych | 67 |
| E. | Wybrane podstawowe operacje administracyjne – czyli co warto wiedzieć, gdy rozpoczynamy pracę na serwerze bazy danych Oracle 10g | 69 |
| F. | Skrypt tworzący zwykłego użytkownika bazy danych Oracle o schemacie zbliżonym do edukacyjnego schematu użytkownika SCOTT proponowanego przez korporację Oracle do nauki podstaw SQL | 81 |

Rozpoczynając temat instalacji narzędzi Oracle, warto wspomnieć, że korporacja Oracle udostępnia m. in. w celu ułatwienia samokształcenia aktualnych i potencjalnych użytkowników jej produktów większość swojego oprogramowania w pełni funkcjonalnej wersji poprzez portal *Oracle Technology Network* (OTN), znajdujący się pod adresem: <http://www.oracle.com/technology/index.html>. Wybierając pakiet instalacyjny należy zwrócić uwagę na wersję systemu operacyjnego, dla którego jest on dedykowany. Zwykle dla 32-bitowych i 64-bitowych systemów operacyjnych przygotowane są różne pakiety instalacyjne. W tym opracowaniu przedstawione zostaną typowe wersje instalacji systemu zarządzania bazą danych, tworzenia bazy danych Oracle i konfiguracji połączenia klienta. Przykładowo pod adresem <http://www.oracle.com/technetwork/database/10201winsoft-095341.html> znajdziemy oprogramowanie Oracle Database 10.2.0.1 pod 32-bitowy system Windows.

Zanim jednak przejdziemy do omówienia poszczególnych przykładów, chcemy krótko zwrócić uwagę na ważne aspekty sprzętowe, decydujące o wydajności i bezpieczeństwie baz danych. Należy podkreślić, że w typowych zastosowaniach kluczowym elementem wpływającym na szybkość działania SZBD Oracle jest wystarczająco duża ilość fizycznej pamięci operacyjnej, a dopiero w drugiej kolejności moc obliczeniowa procesora. Oczywiście, mając na uwadze szybkość dostępu do danych, powinniśmy dysponować dyskami o krótkim czasie dostępu. Natomiast w zastosowaniach produkcyjnych już na poziomie sprzętowym bezwzględnie należy zadbać także o bezpieczeństwo danych, stosując macierze dyskowe (RAID – ang. *redundant array of inexpensive disks*) i awaryjne zasilacze bezprzerwowe (UPS - ang. *uninterruptible power supply*). Tymi podstawowymi wskazówkami należy kierować się, jeśli naszym zadaniem będzie zaprojektowanie konfiguracji sprzętowej serwera, na którym będzie działać baza danych Oracle.

A. Instalacja SZBD Oracle 10g w systemie operacyjnym Windows XP Professional z wybraną opcją tworzenia startowej bazy danych

W tym dodatku przedstawione zostaną kolejne kroki instalacji systemu zarządzania bazą danych Oracle 10g (w wersji 10.2.0.1.0 dla 32-bitowego Windows XP Professional) z wybraną opcją tworzenia w końcowym etapie instalacji startowej bazy danych o domyślnym zestawie plików. Taki typ instalacji można wybrać, gdy nie mamy jeszcze zbyt szerokiej wiedzy na temat struktury plików bazy danych Oracle, a potrzebujemy szybko utworzyć bazę danych na domowym komputerze, np. po to, aby wykonywać ćwiczenia z zakresu języka SQL czy też Oracle PL/SQL.

Krok 1

Zanim rozpoczniemy instalację należy sprawdzić, czy nasz komputer posiada wymagane zasoby sprzętowe. Zgodnie z dokumentacją [1] minimalne wymagania są następujące:

- procesor: przynajmniej **Pentium III 550 MHz** lub kompatybilny (choćaż możliwa powinna być jeszcze instalacja przy procesorze nawet o taktowaniu 300MHz);
- pamięć operacyjna: przynajmniej **512 MB**, chociaż istnieje możliwość instalacji nawet przy **256 MB**, ale wówczas nie można utworzyć bazy danych w trakcie instalacji, a ponadto proces instalacji będzie przebiegał powoli ze względu na intensywne wykorzystanie pamięci wirtualnej dostępnej w systemie operacyjnym;
- pamięć wirtualna (*swap*): wielkości 2-krotnego rozmiaru pamięci operacyjnej
- pamięć dyskowa: **925 MB** (dla SZBD w najbardziej funkcjonalnej wersji Enterprise z dostępnym językiem angielskim i polskim; 905MB wg dokumentacji)
 - + **1046 MB** (dla startowej bazy danych w wersji Enterprise, jeżeli konfiguracja przestrzeni tabel będzie domyślna; 950MB w dokumentacji),
 - + **32 MB** (na dysku systemowym dla instalatora i środowiska JRE – ang. *Java Runtime Environment*)
 - + **125 MB** (w przestrzeni tymczasowej systemu operacyjnego *temp* w pierwszej kolejności poszukiwanej na dysku systemowym)
 - + **3.1 MB** w startowej `C:\Program Files\Oracle`czyli w sumie około: **2.1 GB** (1.94GB wg dokumentacji);
- monitor, potrafiący wyświetlać przynajmniej 256 kolorów.

W praktyce nawet w zwykłych zastosowaniach edukacyjnych, w celu uzyskania odpowiedniego komfortu obsługi zainstalowanego oprogramowania i zadawalającej wydajności działania bazy danych, rekomendowana jest pamięć operacyjna o wielkości przynajmniej 1GB, natomiast rozmiar przestrzeni dyskowej jest ściśle zależny od ilość przechowywanych danych w bazie oraz wybranych komponentów SZBD (jak np. języki narodowe).

Podane powyżej wymagania sprzętowe dotyczą rozważanej tutaj wersji SZBD (*Oracle Database 10g Release 2*). Nowsze wydania (*Oracle Database 11g Release 1* i *Release 2*) mają odpowiednio większe wymagania, które należy sprawdzić w dokumentacji przed rozpoczęciem instalacji.

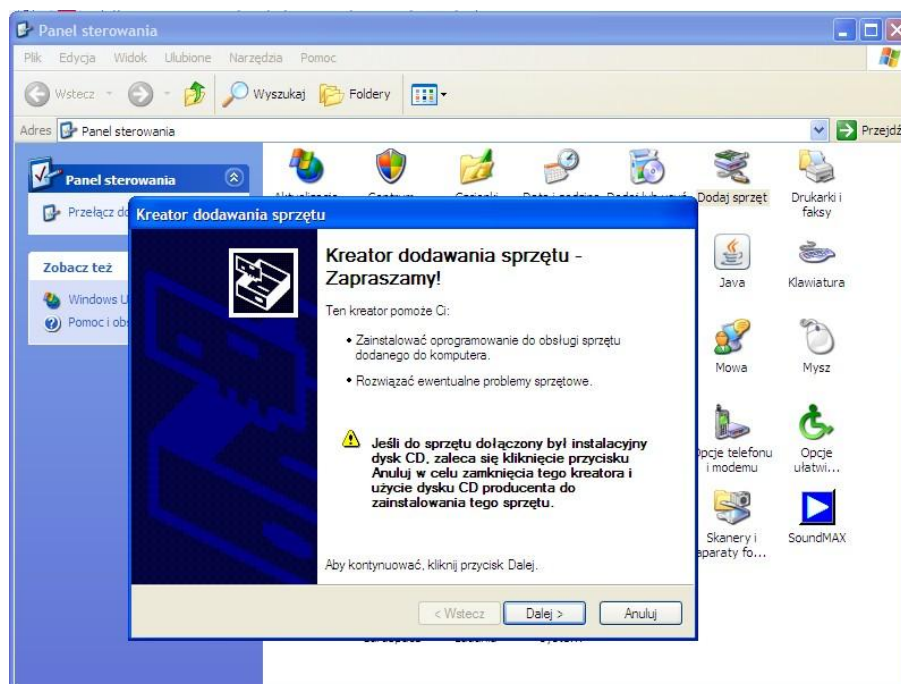
Krok 2

Przed instalacją serwera bazy danych Oracle zaleca się w systemie operacyjnym poprawnie skonfigurowany protokół TCP/IP, który jest podstawowym protokołem komunikacyjnym dla serwerów Oracle poczynając od serwera Oracle 8i. Nie oznacza to konieczności faktycznego połączenia z Internetem lub intranetem. Wystarczy, jeśli komputer będzie posiadał kartę sieciową i skonfigurowany dla niej zgodnie z obowiązującymi regułami protokół TCP/IP, w ramach którego zaleca się również skonfigurowanie serwera systemu nazw domenowych (DNS – ang. *domain name system*) i podanie nazwy domenowej komputera. Poprawne ustawienia DNS znacznie ułatwiają instalację na serwerze usługi tzw. konsoli bazy danych (ang. *database console*), która umożliwia dostęp do wygodnego narzędzia administracyjnego, jakim jest *Oracle Enterprise Manager*, widziany przez użytkownika jako aplikacja przeglądarki internetowej.

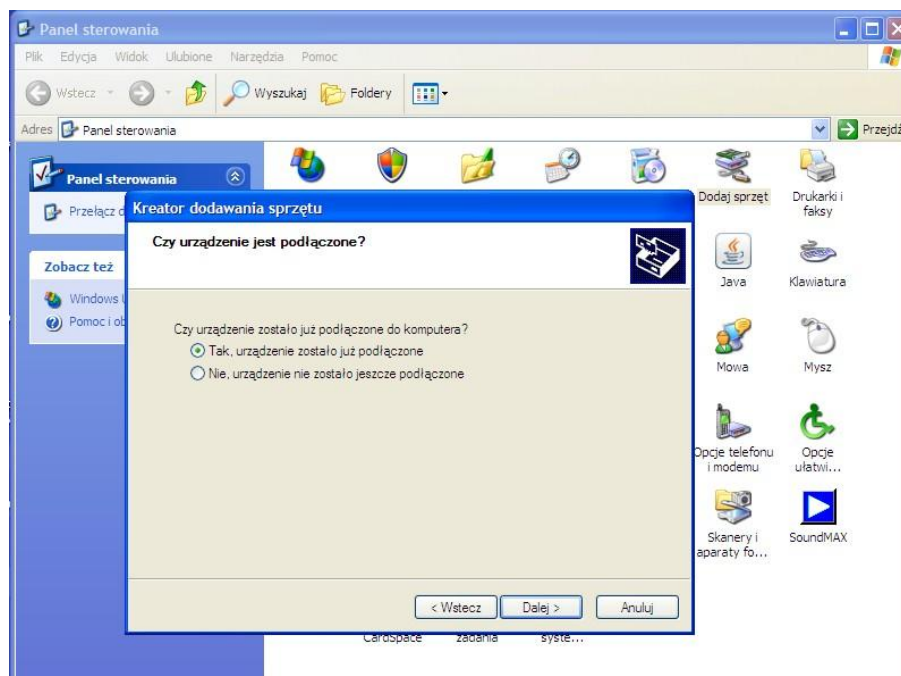
Na koniec tego punktu kierujemy praktyczne wskazówki do użytkowników komputerów łączących się z Internetem za pomocą łącza dynamicznego (poprzez protokół DHCP – ang. *dynamic host configuration protocol*). W takim przypadku, aby uniknąć problemów z konfiguracją usług sieciowych Oracle'a, można wyłączyć połączenie internetowe w systemie operacyjnym i spróbować zainstalować serwer Oracle 10g na tzw. hoście lokalnym (*localhost*), któremu przypisany jest standardowo adres IP: 127.0.0.1. Po zakończeniu instalacji i wystartowaniu bazy danych bez problemu będziemy mogli włączyć połączenie internetowe i korzystać równocześnie z bazy danych pracującej na adresie IP 127.0.0.1 oraz z Internetu dostępnego poprzez inny adres IP przydzielony karcie sieciowej przez protokół DHCP. W tym przypadku należy pamiętać, że analogiczna procedura zwykle będzie konieczna przy każdym starcie bazy danych (tzn. wyłączenie połączenia internetowego, start bazy danych, włączenie połączenia internetowego).

Jednak gdy mamy dostęp do Internetu poprzez DHCP, w celu swobodnego korzystania z serwera bazy danych Oracle włącznie z narzędziem *Oracle Enterprise Manager* zalecane jest wykorzystanie wirtualnej karty sieciowej (*Microsoft Loopback Adapter*). W dalszej części tego dodatku omówiony zostanie właśnie taki sposób instalacji, ponieważ daje on pełną kontrolę nad procesem instalacji w typowych warunkach domowych. Dlatego w tym punkcie pokażemy jak w systemie Windows XP zainstalować wirtualną kartę sieciową i skonfigurować ją tak, aby wykorzystując ją można było w pełni poprawnie zainstalować serwer bazy danych Oracle 10g.

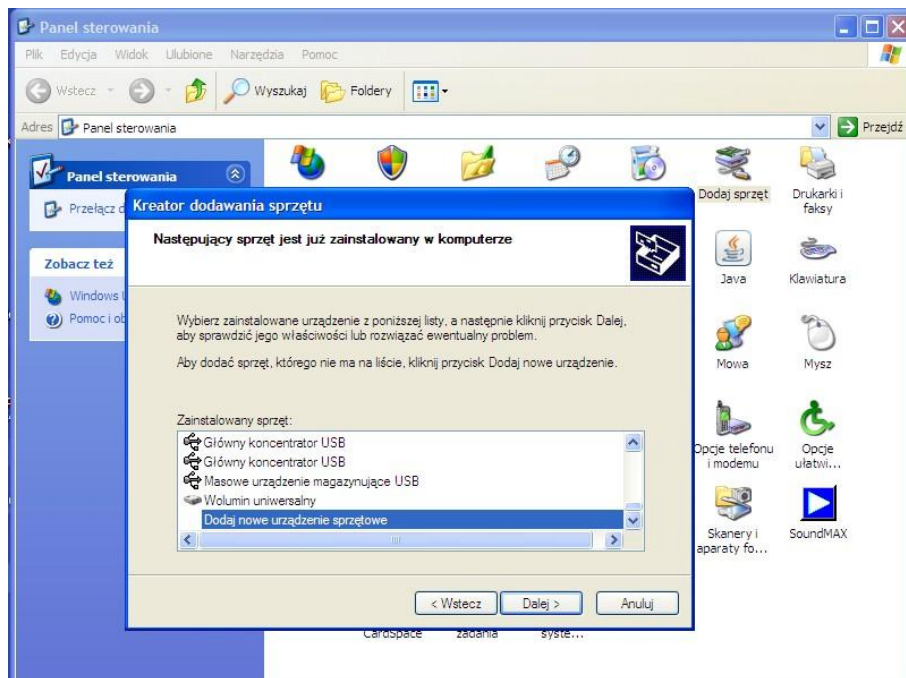
Microsoft Loopback Adapter można doinstalować do systemu operacyjnego poprzez *Kreator dodawania sprzętu* dostępny w *Panelu sterowania* Windows XP, co zostało przedstawione na rys. A.1-A.8.



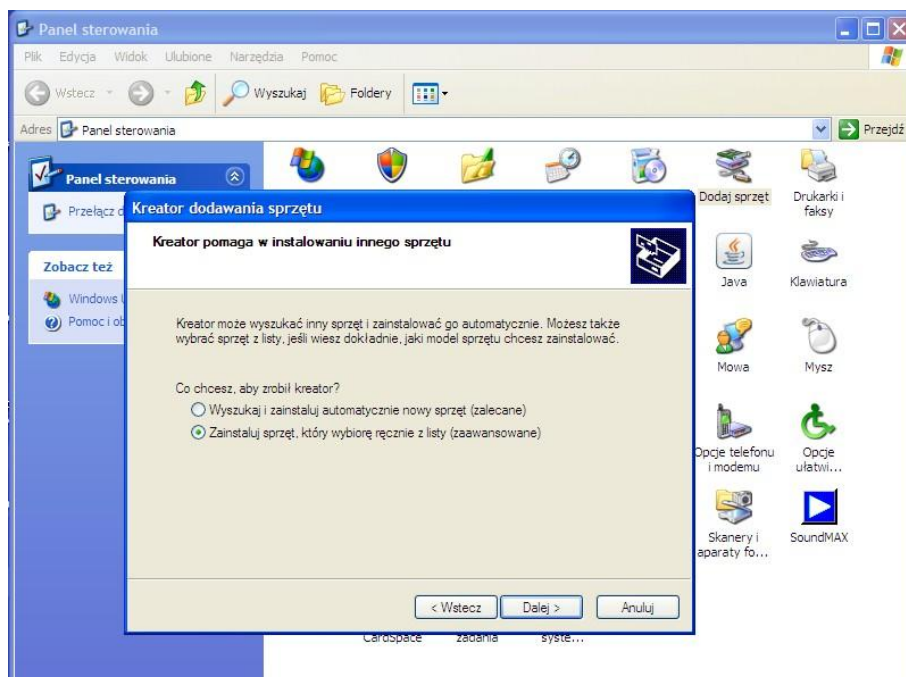
Rys. A.1 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 1)



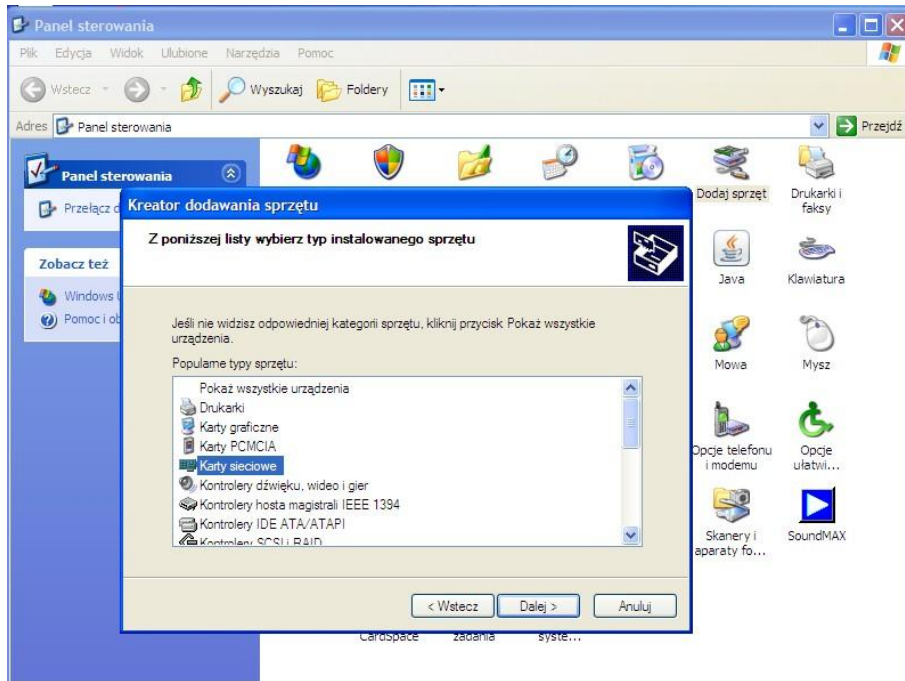
Rys. A.2 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 2)



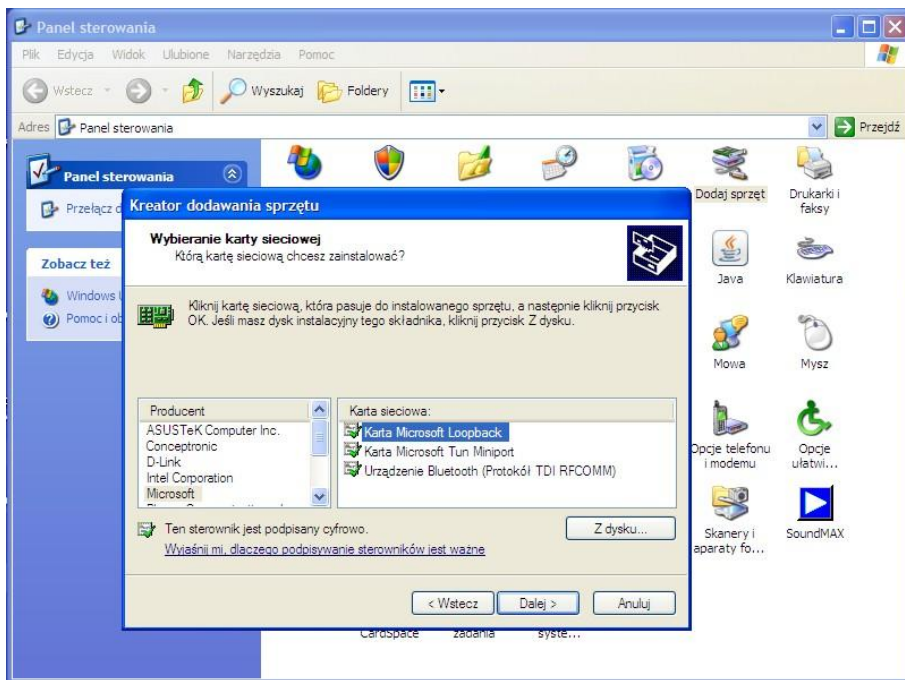
Rys. A.3 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 3)



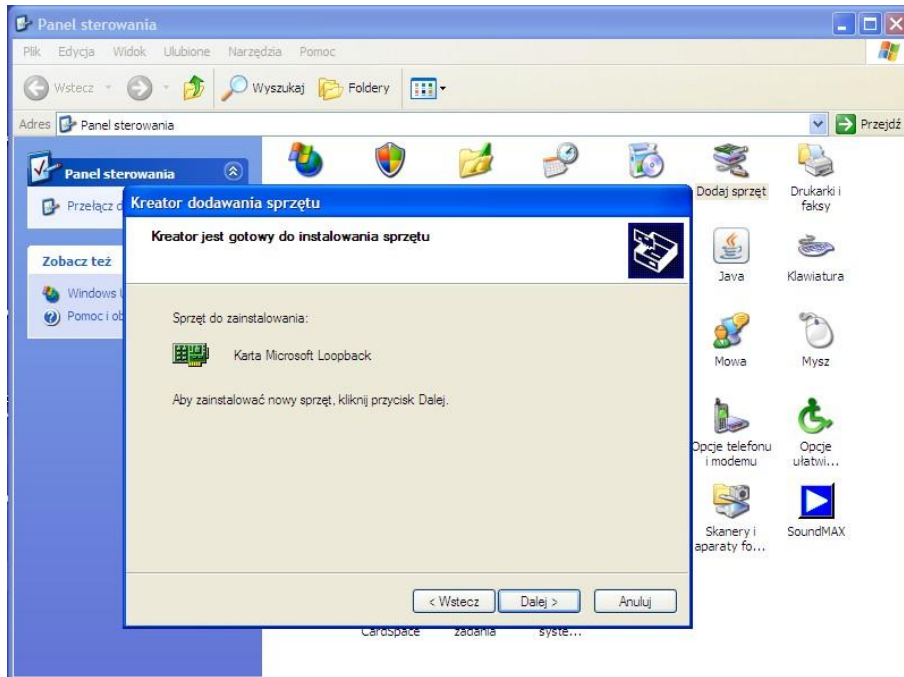
Rys. A.4 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 4)



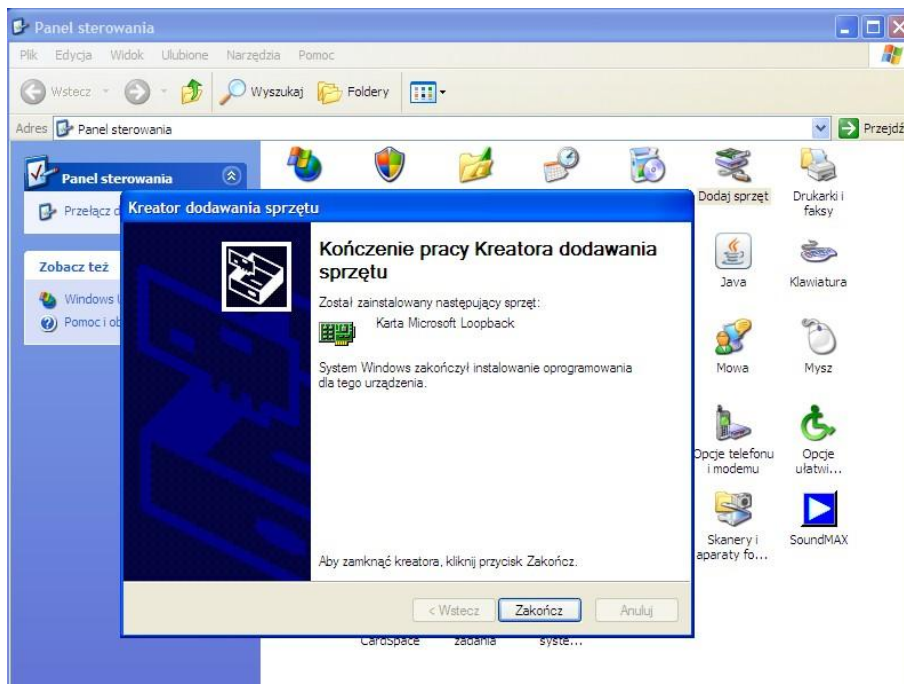
Rys. A.5 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 5)



Rys. A.6 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 6)



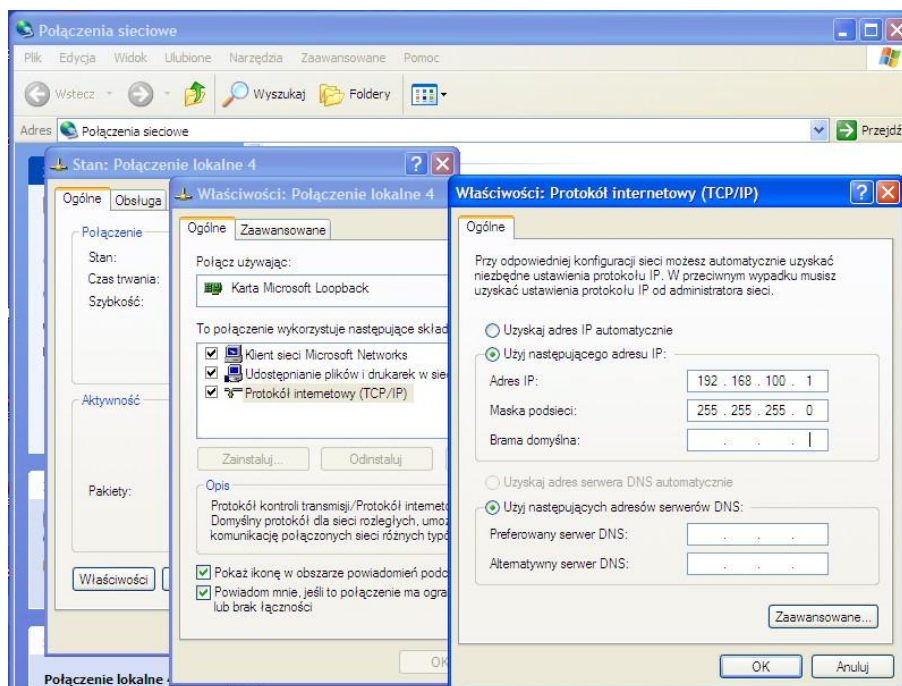
Rys. A.7 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 7)



Rys. A.8 Instalacja *Microsoft Loopback Adapter* w Windows XP (etap 8)

Podczas instalacji wirtualnej karty sieciowej powinno zostać utworzone związane z nią połączenie sieciowe, które należy skonfigurować w sposób prawidłowy dla instalacji serwera bazy danych Oracle. W przedstawionym na rys. A.9 przykładzie jest to *Połączenie sieciowe 4*,

ponieważ w danym systemie operacyjnym były wcześniej utworzone trzy inne połączenia sieciowe.



Rys. A.9 Konfiguracja połączenia sieciowego dla wirtualnej karty sieciowej *Microsoft Loopback Adapter* w Windows XP

W naszym przykładzie protokół TCP/IP został skonfigurowany w jeden ze sposobów zalecanych w dokumentacji Oracle, zgodnie z którą należy przypisać wirtualnej karcie sieciowej jeden z następujących nietrasowalnych (ang. *non-routable*) adresów IP:

- 192.168.x.x (gdzie x jest dowolną wartością pomiędzy 0 a 255)
- 10.10.10.10

przyjmując jako maskę podsieci: 255.255.255.0.

Na koniec należy dodać odpowiedni wpis do pliku `hosts`, dostępnego na dysku systemowym. Jeśli nasz dysk systemowy oznaczony jest literą C, to ten plik konfiguracyjny znajdziemy w lokalizacji `C:\WINDOWS\system32\drivers\etc\hosts`. Dopisujemy tam po wpisie `127.0.0.1 localhost` dotyczącym domyślnego hosta lokalnego, adres IP i nazwę domenową naszego komputera zgodnie ze składnią:

```
adres_IP nazwa_komputera.nazwa_domeny nazwa_komputera
```

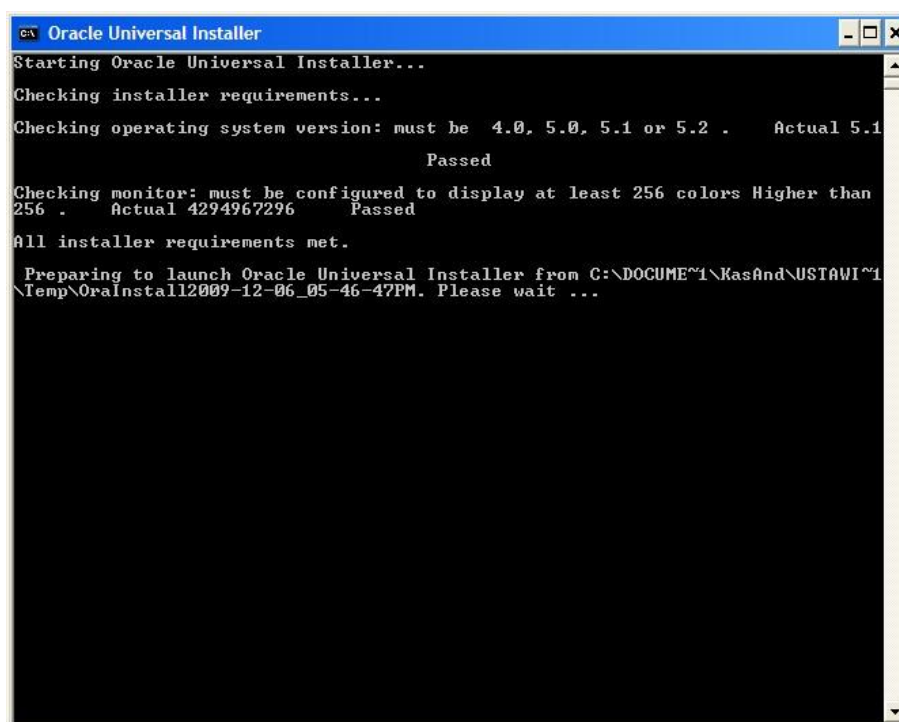
Nazwę domenową komputera możemy sprawdzić w Windows XP w *Panelu sterowania* w grupie *Wydajność i konserwacja* w części *System*, gdzie wybieramy zakładkę *Nazwa komputera*, gdzie szczegóły dotyczące domeny znajdziemy po wybraniu opcji *Zmień*. Dla potrzeb naszej przykładowej instalacji w pliku `hosts` dokonano poniższy wpis:

```
192.168.100.1          BMW.          BMW
```

ponieważ dotyczy on komputera o nazwie BMW, który nie należy do żadnej domeny, a jedynie do grupy roboczej.

Krok 3

Logujemy się do systemu operacyjnego na konto użytkownika o uprawnieniach administracyjnych. Jeżeli instalator nie rozpocznie działania automatycznie po włożeniu pierwszej instalacyjnej płyty CD, to uruchamiamy program `setup.exe`, znajdujący się na tej płycie w katalogu głównym. Warto podkreślić, że program ten spełni swoje zadanie równie dobrze po przegraniu zawartości płyt instalacyjnych na dysk twardy lub po rozpakowaniu pakietu instalacyjnego pobranego w formie skompresowanej z portalu OTN.



```
Oracle Universal Installer
Starting Oracle Universal Installer...
Checking installer requirements...
Checking operating system version: must be 4.0, 5.0, 5.1 or 5.2 .   Actual 5.1
                                     Passed
Checking monitor: must be configured to display at least 256 colors Higher than
256 .   Actual 4294967296   Passed
All installer requirements met.
Preparing to launch Oracle Universal Installer from C:\DOCUME~1\KasAnd\USTAWI~1
\Temp\OraInstall12009-12-06_05-46-47PM. Please wait ...
```

Rys. A.10 Sprawdzenie wymagań systemowo-sprzętowych przez program instalacyjny

Pierwszym zadaniem, które wykonuje instalator jest sprawdzenie (rys. A.10), czy wymagane elementy konfiguracji systemowo-sprzętowej są dostępne, w tym przypadku są to odpowiednia

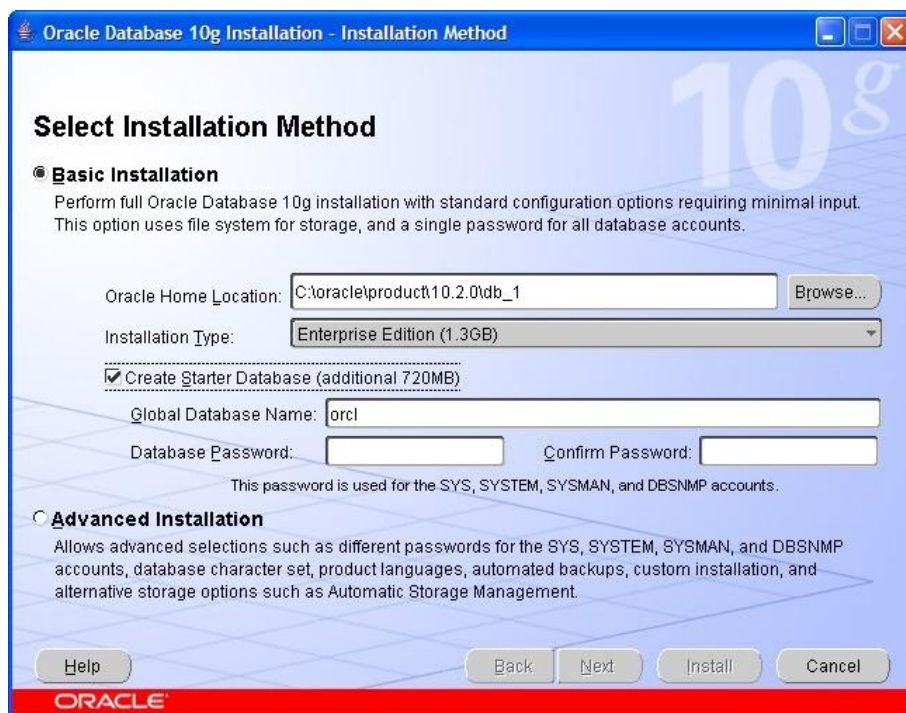
wersja systemu operacyjnego i dostępna liczba kolorów monitora. Jeśli dowolny z tych elementów nie jest właściwy, to instalator wskazuje problem i uniemożliwia kontynuowanie instalacji. Niekiedy mogą pojawić się także problemy z niesprawdzanymi przez ten instalator ważnymi parametrami systemowo-sprzętowymi, np. takimi jak rozmiar pamięci wirtualnej (*swap*) i przestrzeni tymczasowej (*temp*), które są sprawdzane np. przez instalator pakietu projektancko-programistycznego (Oracle Developer Suite 10g) oraz w przypadku instalacji SZBD Oracle 10g w niektórych wersjach Linuxa. W większości przypadków, w których pojawiają się tego typu problemy (niezależnie od tego, czy są sygnalizowane odpowiednio wcześniej przez instalator czy też dopiero w momencie wystąpienia błędu programu instalującego), jeżeli na przeszkodzie nie stoją ograniczenia sprzętowe, możemy dokonać koniecznych zmian ustawień w systemie operacyjnym, np. zmienić sterownik monitora lub powiększyć rozmiar pamięci wirtualnej czy przestrzeni tymczasowej. Najczęściej spotykanym problemem jest zbyt mały obszar dysku zarezerwowany na pamięć wirtualną. Ten parametr możemy zmienić w systemie Windows XP w *Panelu sterowania* w części *Wydajność i konserwacja* w grupie narzędzi *System*, gdzie w zakładce *Zawansowane* wybieramy przycisk *Ustawienia* w opcji *Wydajność* i następnie w znajdującej się tam zakładce *Zawansowane* odszukujemy opcję *Pamięć wirtualna* z przyciskiem *Zmień*, którego naciśnięcie udostępnia panel zmian parametrów pliku stronicowania, pełniącego rolę pamięci wirtualnej. Kończąc ten punkt wart zwrócić uwagę na fakt, że rozmiar pamięci wirtualnej nabiera szczególnego znaczenia i może wymagać zwiększenia, gdy komputer posiada mało fizycznej pamięci operacyjnej, tzn. poniżej 512MB.

Do wykorzystywanej w rozpatrywanym przykładzie wersji programu instalacyjnego i wymagań instalacyjnych wrócimy jeszcze w kroku 6 opisu procedury instalacyjnej.

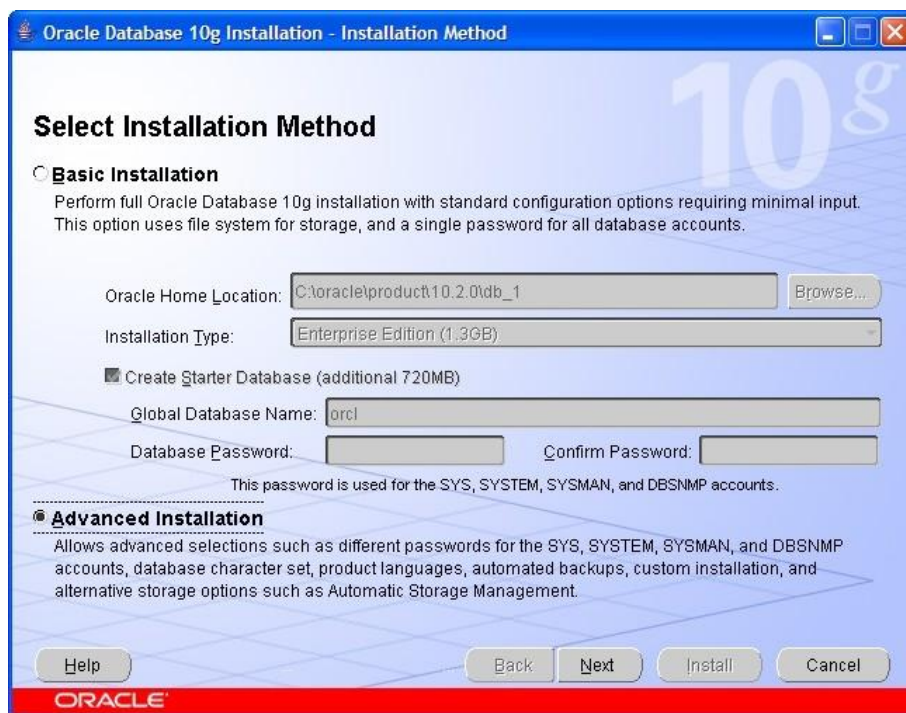
Krok 4

Następnie rozpoczyna się właściwa instalacja SZBD z ewentualnym tworzeniem bazy danych. W pierwszym etapie mamy możliwość wyboru metody instalacji: standardowej (rys. A.2) bądź zaawansowanej (rys. A.3). My wybierzemy instalację zaawansowaną, która pozwala na pełną kontrolę nad parametrami instalacji i daje swobodną możliwość ustawienie ich w zależności od potrzeb. Zauważmy tylko w związku ze standardową metodą instalacji, że chcąc mieć dostęp do pełnej funkcjonalności instalowanego pakietu, należy wybrać jego wersję Enterprise (jak pokazano to na rys. A.2). Natomiast zycząc sobie utworzenia startowej bazy danych w końcowym etapie instalacji, musimy wybrać opcję *Create Starter Database*. Przy

czym warto zauważyć, że rozmiar startowej bazy danych jest w rzeczywistości większy niż podaje instalator i wynosi przynajmniej 950MB.



Rys. A.11 Wybór standardowej metody instalacji SZBD Oracle 10g.

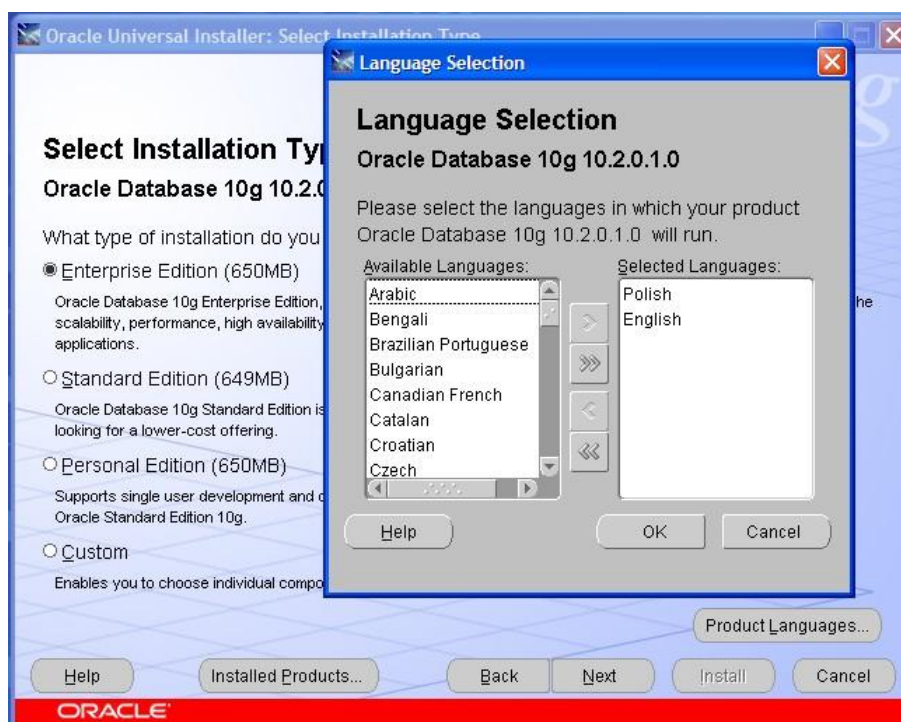


Rys. A.12 Wybór zaawansowanej metody instalacji SZBD Oracle 10g

Krok 5

W ramach instalacji zaawansowanej w jej początkowym etapie możemy samodzielnie ustawić następujące parametry:

1. Wybrać, jakie języki narodowe mają zostać zaimplementowane w instalowanym pakiecie.



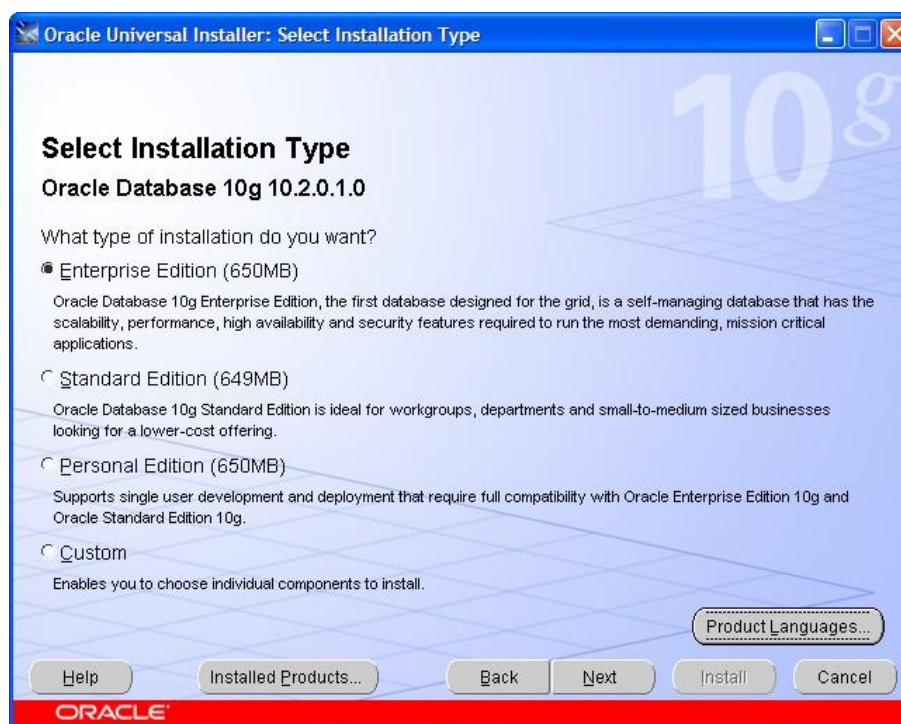
Rys. A.13 Wybór języków narodowych dla instalowanego pakietu

Widoczny na rys. A.13 wybór języka angielskiego jest domyślnie proponowany przez instalator, podobnie jest z językiem polskim, który w polskojęzycznej wersji systemu operacyjnego jest domyślnie dołączony do języka angielskiego. Warto wiedzieć, że nie wszystko (w tym również program instalacyjny) zostało przetłumaczone na języki narodowe, dlatego dobrze jest pozostawić wybór języka angielskiego, który bez konfliktów koegzystuje z innymi językami narodowymi.

2. Wybrać typ instalowanego pakietu

Aby zapewnić dostęp do pełnej funkcjonalności SZBD i działających pod jego kierunkiem baz danych, należy wybrać wersję Enterprise instalowanego pakietu (rys. A.14).

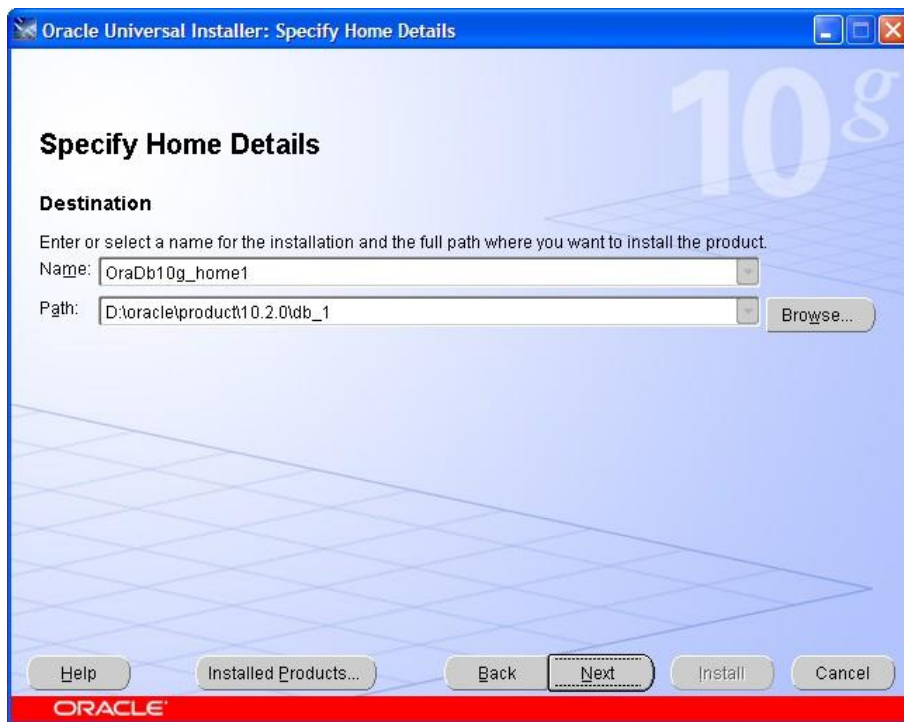
Przy czym warto zauważyć, że rozmiar przestrzeni dyskowej zajmowanej przez SZBD jest w rzeczywistości większy niż podaje instalator i wynosi przynajmniej 905MB.



Rys. A.14 Wybór typu instalowanego pakietu

- Wybrać ścieżkę dostępu, w której ma być zainstalowany SZBD Oracle 10g, określaną w nomenklaturze Oracle jako `ORACLE_HOME` oraz jej nazwę (`ORACLE_HOME_NAME`), dodatkowo charakteryzującą w rejestrach systemu operacyjnego Windows tę ścieżkę dostępu.

Na rys. A.15 są to odpowiednio `D:\oracle\product\10.2.0\db_1` oraz `OraDb10g_home1`. Oczywiście lokalizację SZBD należy wybrać w miarę możliwości nie na dysku systemowym, chociaż instalator domyślnie sugeruje właśnie taką opcję. Natomiast zaproponowanej dla niej nazwy `ORACLE_HOME_NAME` raczej nie należy zmieniać. Ostatnia uwaga wiąże się z tym, że w danym systemie operacyjnym może zostać zainstalowanych więcej pakietów narzędzi Oracle. Każdy z nich powinien mieć inną lokalizację, co wiąże się z innym nazwą domu Oracle, która jest dobierana automatycznie przez instalator tak, aby uniknąć kolizji z poprzednio zainstalowanymi produktami.



Rys. A.15 Wybór tzw. nazwy domu Oracle (opcja *Name*) i przypisanie jej lokalizacji SZBD (opcja *Path*), czyli odpowiednio ORACLE_HOME_NAME i ORACLE_HOME

Krok 6

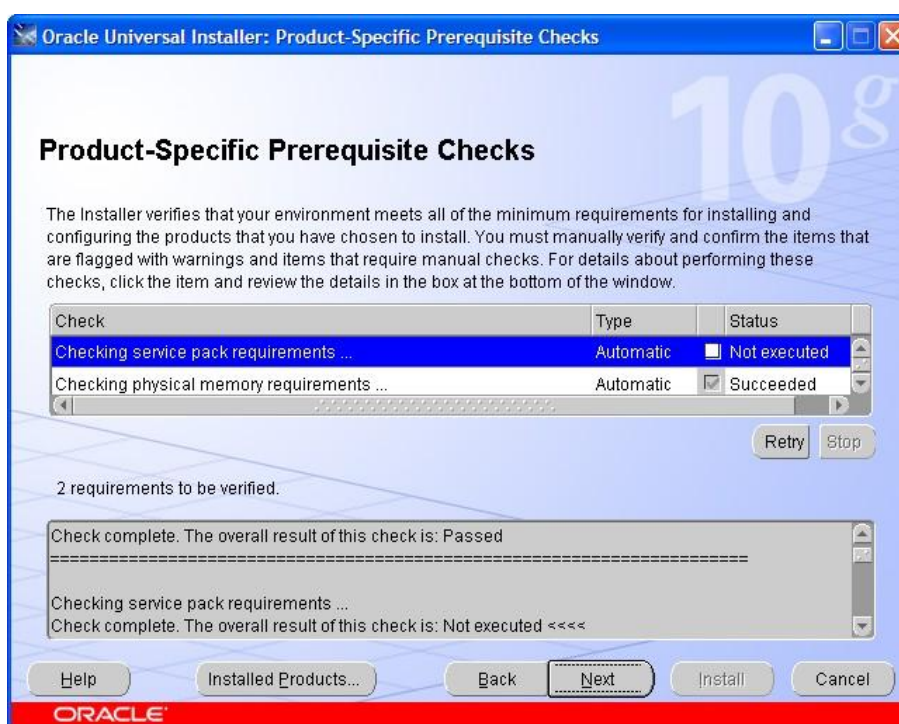
Po wybraniu tych wstępnych parametrów program instalacyjny przechodzi do kontynuacji weryfikacji wymagań systemowo-sprzętowych (rys. A.16), rozpoczętej na początku instalacji i opisanej w kroku 3. W tym miejscu sprawdzane są kolejno:

- wersja systemu operacyjnego z dokładnością do jego Service Packa (tu warto wspomnieć, że każda wersja Windows XP Professional jest akceptowana)
- rozmiar dostępnej fizycznej pamięci operacyjnej
- konfiguracja sieciowa
- weryfikacja lokalizacji określanej jako ORACLE_BASE (jeśli została ustawiona);
- weryfikacja dostępnego miejsca i poprawności lokalizacji określanej jako ORACLE_HOME
- weryfikacja możliwych niezgodności innych ustawień systemu operacyjnego, w szczególności w odniesieniu do ewentualnych poprzednio przeprowadzonych w nim instalacji narzędzi Oracle (w tym zawartości ewentualnie niepustej lokalizacji podanej w ORACLE_HOME).

Omówiony w poprzednim kroku parametr ORACLE_HOME, jak i ORACLE_BASE, który zawiera propozycję nadrzędnej ścieżki dostępu do zainstalowanych narzędzi Oracle i w naszym

przykładzie ma przypisaną ścieżkę D:\oracle\product\10.2.0, mogą być zmiennymi środowiskowymi systemu operacyjnego, a w systemie Windows zostają wpisane do rejestru w kluczy HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_ORACLE_HOME_NAME, gdzie w naszym przykładzie ORACLE_HOME_NAME = OraDb10g_home1.

Zalecane jest dokładne sprawdzenie wyników tego testu, chociaż zwykle możemy kontynuować instalację, nawet jeśli nie wszystkie punkty testu zakończyły się sukcesem. Aby to uwidocznić, wydruk wyników testu poniżej rys. A.16 dotyczy uruchomienia instalatora bez wykonania kroku 2 (tzn. bez zainstalowania i konfiguracji wirtualnej karty sieciowej) na komputerze z włączonym połączeniem sieciowym działającym w oparciu o protokół DHCP.



Rys. A.16 Kontynuacja weryfikacji wymagań systemowo-sprzętowych.

Poniżej wydruk wyników wykonanego testu.

```
Checking operating system requirements ...
```

```
Expected result: One of 4.0,4.1,5.0,5.1,5.2
```

```
Actual Result: 5.1
```

```
Check complete. The overall result of this check is:
```

```
Passed
```

```
Checking service pack requirements ...
```

```
Check complete. The overall result of this check is: Not  
executed <<<<
```

```
OUI-15038: Unable to execute rule 'CheckServicePack'  
from ruleset 'ServicePackChecks'. The underlying test
```

'oracle.sysman.oip.oipc.oipcz.OipczServicePackChecks.ckServicePack' has thrown an unhandled exception [For input string: "Pac"]. If the exception results in a failure of the test and the exception needs to be propagated, then encapsulate the exception in the result object and return the result.

Recommendation: Install the recommended Service Pack.

=====
Checking physical memory requirements ...

Expected result: 256MB
Actual Result: 1015MB
Check complete. The overall result of this check is:
Passed

=====
Checking Network Configuration requirements ...

Check complete. The overall result of this check is: **Not executed** <<<<
Recommendation: Oracle supports installations on systems with DHCP-assigned IP addresses; However, before you can do this, you must configure the Microsoft LoopBack Adapter to be the primary network adapter on the system. See the Installation Guide for more details on installing the software on systems configured with DHCP.

=====
Validating ORACLE_BASE location (if set) ...

Check complete. The overall result of this check is:
Passed

=====
Checking Oracle Home path for spaces...

Check complete. The overall result of this check is:
Passed

=====
Checking Oracle Home path for location ...

Check complete. The overall result of this check is:
Passed

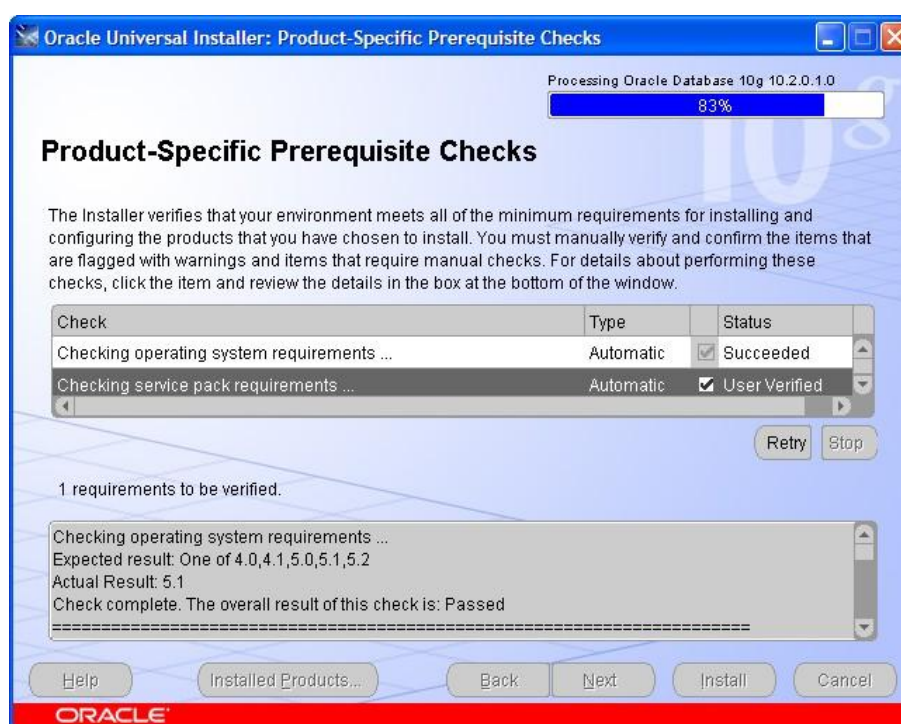
=====
Checking for proper system clean-up....

Check complete. The overall result of this check is:
Passed

=====
Checking for Oracle Home incompatibilities

Actual Result: NEW_HOME
Check complete. The overall result of this check is:
Passed

Na przedstawionym wydruku wyniku testu widzimy, że nie powiodło się sprawdzenie wersji Service Packa systemu operacyjnego oraz konfiguracji sieciowej. Pierwszy problem możemy zignorować, ponieważ Windows XP Professional niezależnie od wersji Service Packa jest odpowiednim systemem operacyjnym do instalacji Oracle Database 10g Release 2, natomiast koniecznie musimy dokładnie przyjrzeć się niewłaściwej konfiguracji sieciowej. W tym przypadku program instalacyjny radzi nam skonfigurować wirtualną kartę sieciową *Microsoft LoopBack Adapter* jako główne łącze sieciowe w systemie. W praktyce wystarczy skonfigurować wirtualną kartę sieciową, która niekoniecznie musi być urządzeniem podstawowego połączenia sieciowego. W związku z tym wykonujemy procedurę opisaną w kroku 2 i powtarzamy weryfikację wymagań, zaznaczając pola wyboru przy testach, które nie przeszły pomyślnie, a które zamierzamy powtórzyć, i naciskamy przycisk *Retry*. W rezultacie otrzymujemy powtórkę błędu testu Service Packa, ale konfiguracja sieciowa zostaje pomyślnie zweryfikowana, dlatego decydujemy się przejść do następnego kroku instalacji (rys. A.17).



Rys. A.17 Przejście do następnego kroku instalacji po powtórce niepomyślnie zweryfikowanych testów wymagań systemowo-sprzętowych. Poniżej wydruk wyników powtórzonych testów.

```
Check complete. The overall result of this
check is: Not executed <<<<
```

```
OUI-15038: Unable to execute rule
'CheckServicePack' from ruleset
'ServicePackChecks'. The underlying test
'oracle.sysman.oip.oipc.oipcz.OipczServicePackC
hecks.checkServicePack' has thrown an unhandled
exception [For input string: "Pac"]. If the
exception results in a failure of the test and
the exception needs to be propagated, then
encapsulate the exception in the result object
and return the result.
```

```
Recommendation: Install the recommended Service
Pack.
```

```
=====
```

```
Checking Network Configuration requirements ...
```

```
Check complete. The overall result of this
check is: Passed
```

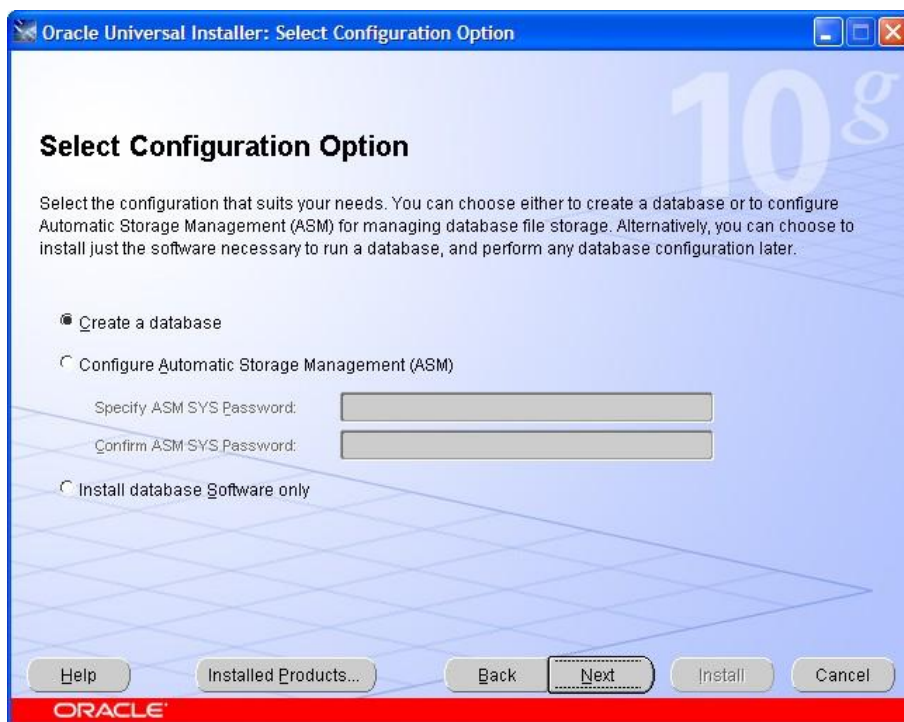
```
=====
```

W praktyce zdarza się podejmować decyzję o kontynuowaniu instalacji z nie w pełni zweryfikowanymi wymaganiami systemowo-sprzętowymi. Jednak należy tego unikać, a jeśli się na to zdecydujemy, to powinniśmy to robić świadomie, mając na uwadze możliwe negatywne konsekwencje dla składowanych w bazie danych. Przykład przedstawiony na rys. A.17 i znajdujący się poniżej niego wydruk pokazują sytuację, którą można zaliczyć do dopuszczalnych, ponieważ zgodnie z dokumentacją wersja Service Packa systemu Windows XP Professional nie ma znaczenia dla instalacji SZBD i tworzenia bazy danych Oracle 10g Release 2. Natomiast konfiguracja sieciowa, którą poprawiliśmy zgodnie z wymaganiami, jest niewątpliwie elementem dla sprawnego i w pełni funkcjonalnego działania bazy danych Oracle i nie należy jej lekceważyć w procesie instalacji.

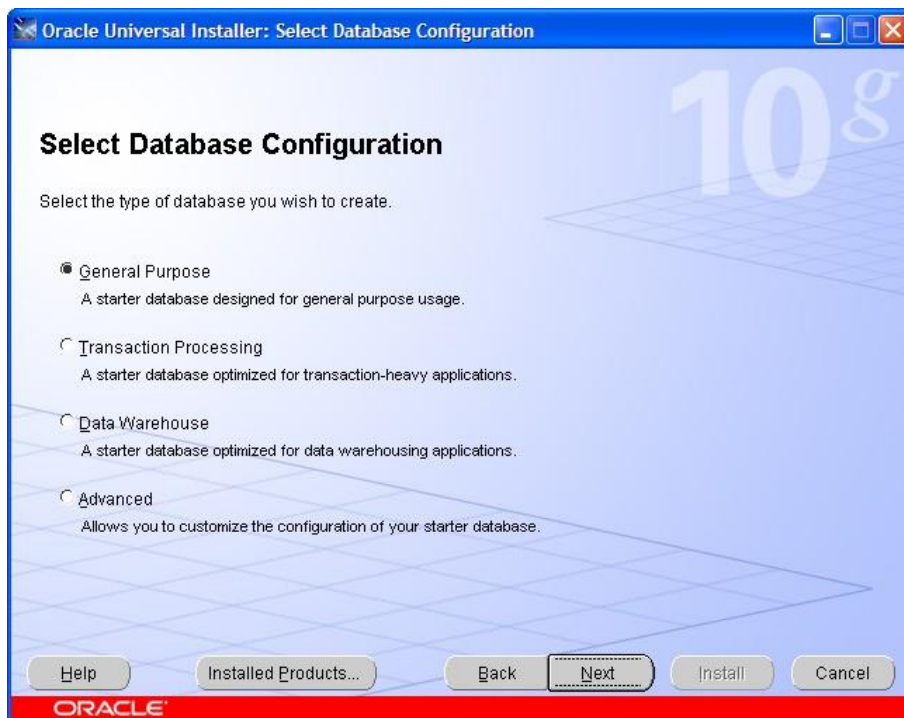
Krok 7

Teraz decydujemy (rys. A.18), czy startowa baza danych ma być utworzona w końcowym etapie instalacji (opcja *Create a database*) czy też nie (opcja *Install database Software only*). Jeśli tak, to dalej możemy ustawić wybrane parametry bazy danych w zależności od jej przeznaczenia i przewidywanych potrzeb.

Przed wszystkim musimy wskazać, do jakich celów ma służyć nasza baza (rys. A.19). Czy ma być to baza do ogólnych zastosowań (opcja *General Purpose*), dla systemu transakcyjnego (opcja *Transaction Processing*), dla hurtowni danych (opcja *Data Warehouse*), czy też której podstawowe parametry skonfigurujemy samodzielnie (opcja *Advanced*).

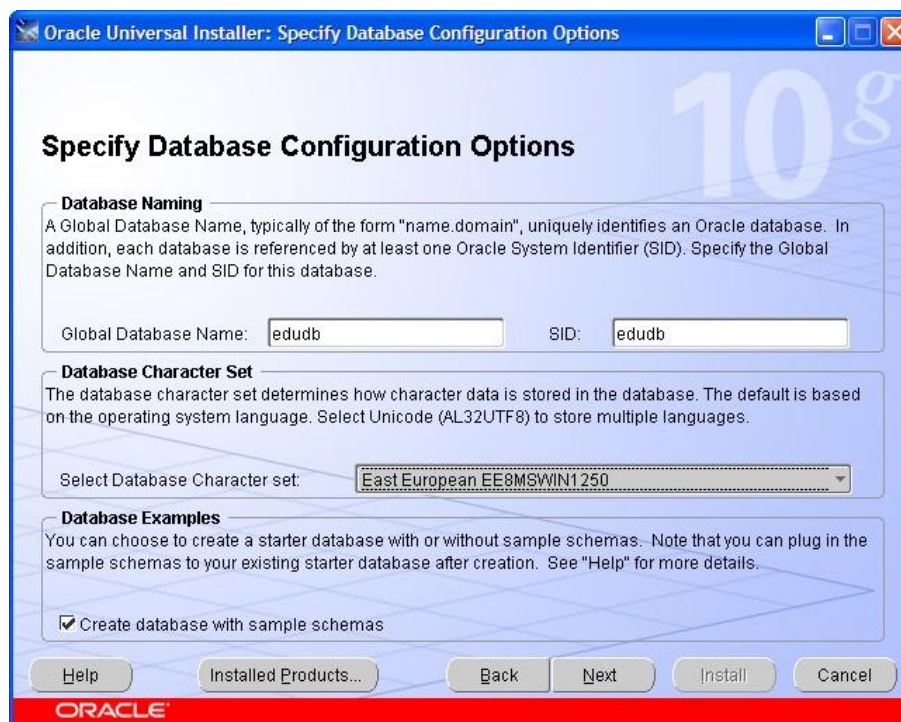


A.18 Wybór czy startowa baza danych ma być utworzona w końcowym etapie instalacji



A.19 Wybór typu konfiguracji bazy danych w zależności od jej przeznaczenia

Ponieważ zakładamy, że niniejszy dodatek ma służyć pomocą podczas instalacji bazy na domowym komputerze, która będzie służyć do wykonywania ćwiczenia z zakresu języka SQL czy też Oracle PL/SQL, dlatego w naszym przykładzie wybieramy (rys. A.19) instalację bazy danych do ogólnych zastosowań. Taki wybór będzie wygodny i całkowicie wystarczający dla wszystkich osób, które dopiero zaczynają poznawać serwer bazy danych Oracle.

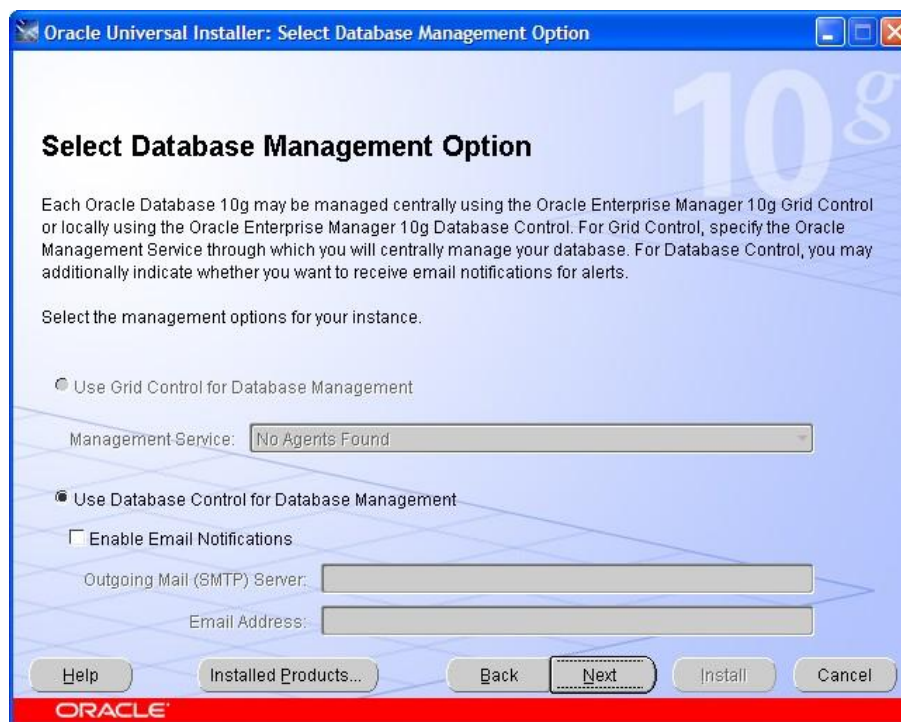


A.20 Wybór nazwy i identyfikatora bazy oraz jej zestawu znaków

Następnie podajemy bardzo ważne parametry bazy danych (rys. A.20), którymi są:

- globalna nazwa bazy danych (ang. *Global Database Name*), w ogólności mogąca posiadać składnię domenową (np. oracle.us.edu.pl); jednak w naszym przypadku nie określiliśmy domeny w systemie operacyjnym, dlatego było zasadne ustalenie jednoczłonowej nazwy bazy danych;
- SID (ang. System ID), unikalny identyfikator naszej bazy danych, który zwykle (choć nie obowiązkowo) jest identyczny z pierwszym członem domenowej globalnej nazwy bazy danych;
- zestaw znaków bazy danych (ang. Database Character Set), nad którym należy dobrze zastanowić się, ponieważ jego zmiana po zainstalowaniu bazy danych nie jest łatwą procedurą, która grozi nawet utratą pewnych danych, a niekiedy nie jest możliwa do wykonania w zależności od pierwotnego i oczekiwanego docelowego zestawu znaków.

Ponadto możemy wybrać, czy mają zostać utworzone przykładowe schematy edukacyjne, np. popularnego użytkownika SCOTT. Oczywiście w naszym edukacyjnym przykładzie decydujemy się na zainstalowanie tych schematów (opcja *Create database with sample schemas*).

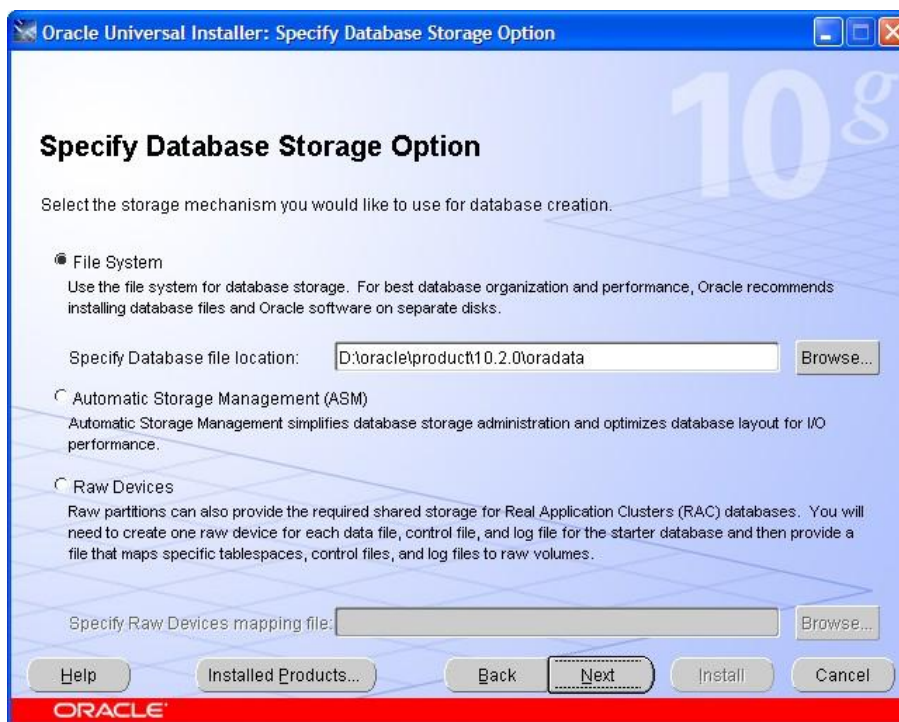


A.21 Wybór sposobu zarządzania bazą danych

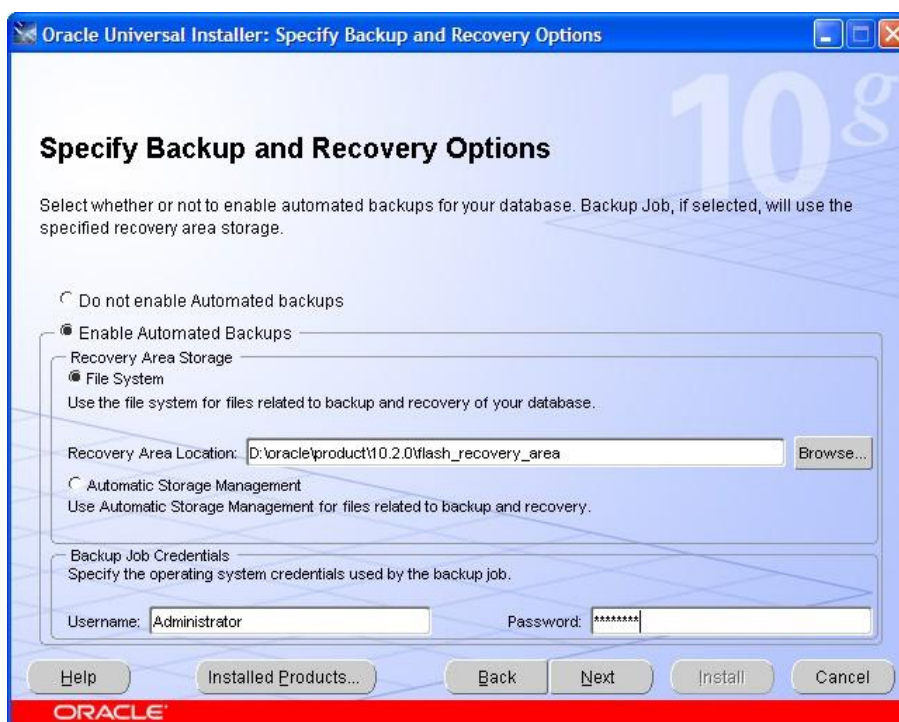
Jak widać na rys. A.21 spośród dwóch standardowych sposobów zarządzania bazą Oracle 10g mamy dostępne jedynie zastosowanie dostępnych mechanizmów instalowanej bazy danych (opcja *Use Database Control for Database Management*), natomiast nie jest aktywna opcja *Use Grid Control for Database Management*, która wskazywałaby, że baza byłaby włączona do pewnego klastra baz i w ramach niego zarządzana. Jest to spowodowane faktem prowadzenia instalacji na wirtualnej karcie sieciowej, która została rozpoznana przez instalator.

Kolejną bardzo ważną własnością bazy danych jest rodzaj urządzeń, które zostaną przeznaczone do składowania danych. Możemy w tym zakresie skorzystać ze zwykłego systemu plików, z grupy dysków sterowanej przez zautomatyzowany mechanizm zarządzania (ASM - ang. *Automatic Storage Management*) oraz z urządzeń surowych (ang. *raw devices*). Choć baza Oracle jest bardziej wydajna w dwóch ostatnich przypadkach, to w warunkach nieprodukcyjnych wybieramy składowanie danych w systemie plików (jak pokazano to na rys. A.22). Następnie możemy już na etapie instalacji zdecydować się na to, aby SZBD

automatycznie tworzył kopie zapasowe naszej bazy. Pokazano to na rys. A.23, gdzie widać, że podobnie jak w przypadku składowania danych naszej bazy, także ich kopie zapasowe będą przechowywane w systemie plików, w lokalizacji podanej w polu *Recovery Area Location*.



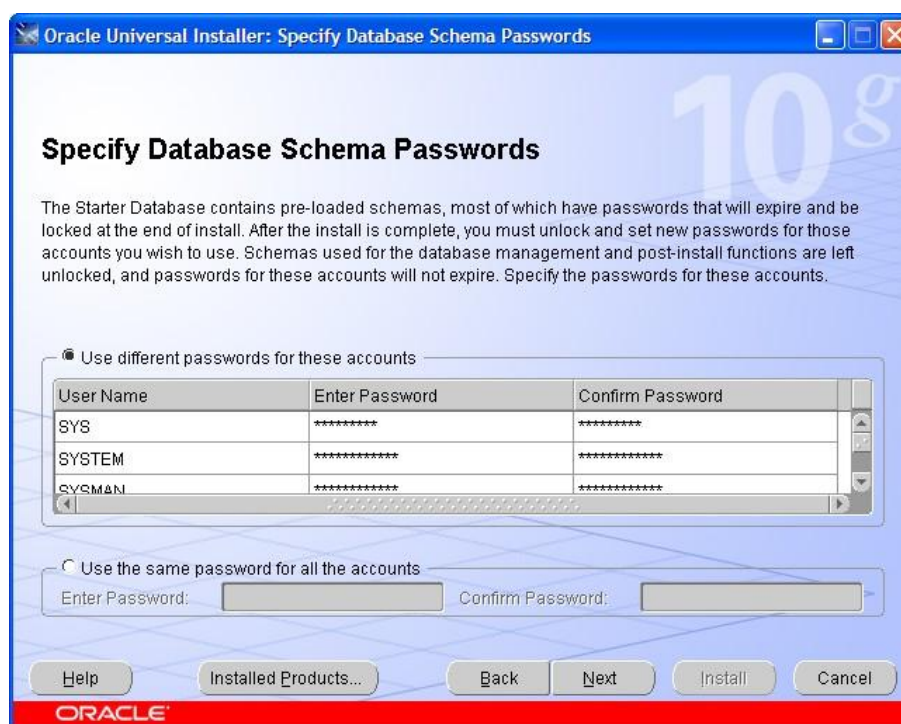
A.22 Wybór rodzaju urządzeń przeznaczonych do składowania danych



A.23 Wybór zautomatyzowanego procesu tworzenia kopii zapasowych i jego parametrów

W przypadku zautomatyzowanego tworzenia kopii zapasowych istotne jest podanie użytkownika systemu operacyjnego wystarczająco uwierzytelnionego do tego zadania oraz jego hasła (w opcji *Backup Job Credentials* pola *Username* i *Password*), ponieważ program Oracle Recovery Manager (Oracle RMAN) wykonujący taką kopię zapasową musi zalogować się na użytkownika systemu operacyjnego mającego odpowiednie uprawnienia (np. odpowiedni dostęp do dysku), będącego jednocześnie systemowym administratorem bazy danych Oracle. W przypadku Windows możemy w tym celu wybrać użytkownika Administrator, ponieważ przeprowadzamy instalację serwera bazy danych Oracle na jego koncie. Natomiast w Linuxie użytkownikami systemu operacyjnego odpowiednimi do tworzenia kopii zapasowych bazy danych, będą ci, którzy należą do grupy dba, jak np. użytkownik oracle.

Ostatnim parametrem, który pozostaje do ustalenia są hasła dla czterech (niesystemowych) użytkowników administracyjnych bazy danych Oracle 10g, tj. SYS, SYSTEM, SYSMAN oraz DBSNMP, który nie jest widoczny na rys. A.24. Z powodów bezpieczeństwa dobrze jest wybierać inne hasło dla każdego z tych użytkowników.



A.24 Wybór haseł dla użytkowników administracyjnych bazy danych Oracle 10g

Krok 8

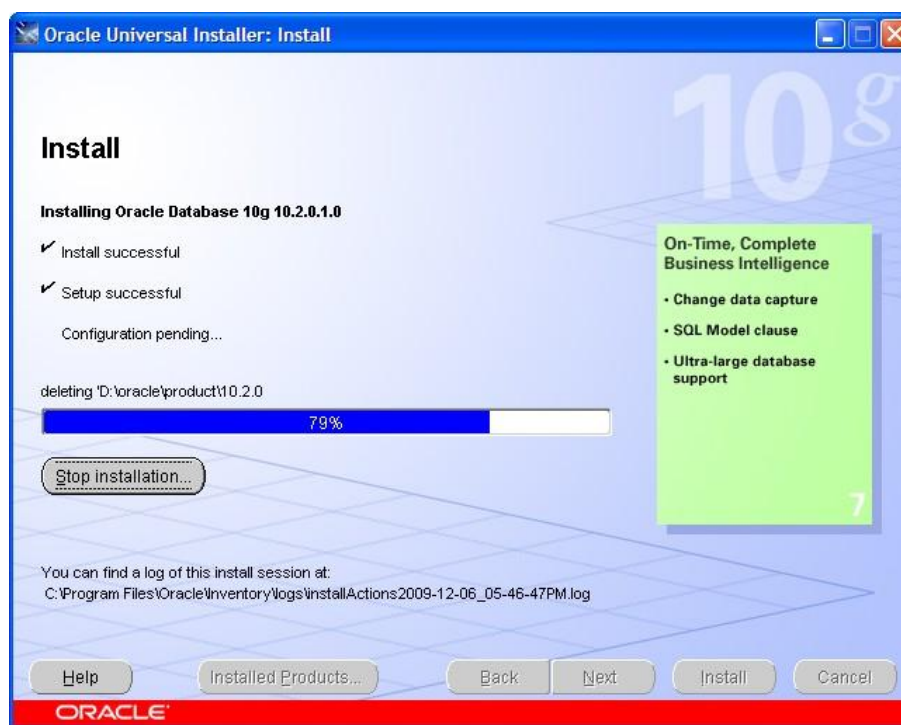
Po tym jak podjęliśmy już wszystkie decyzje dotyczące parametrów instalacji możemy uruchomić proces instalacji, mając możliwość wcześniejszej weryfikacji ustalonych parametrów i wybranych pakietów oprogramowania (rys. A.25).



A.25 Weryfikacja parametrów instalacji i narzędzi, które będą instalowane

Najpierw instalowane jest oprogramowanie serwera. Szczegółowa informacja o przeprowadzonej instalacji jest zapisywana w katalogu `C:\Program Files\Oracle\Inventory\logs` do odpowiedniego tekstowego pliku dziennika, którego nazwa zaczyna się od `installActions` i zawiera datę i godzinę rozpoczęcia instalacji (rys. A.26). W przypadku wystąpienia błędów lub ostrzeżeń podczas instalacji właśnie w tym pliku można znaleźć opis problemów zidentyfikowanych przez program instalacyjny. Warto w tym miejscu zauważyć, że podczas procesu instalacji SZBD Oracle mogą pojawić się pewne błędy lub ostrzeżenia, pomimo że postaraliśmy się spełnić wszystkie wymagania opisane w kroku 3 i 6 procedury instalacyjnej. Wówczas zgodnie komunikatem instalatora należy, jeśli to możliwe, zrobić odpowiednie zmiany w systemie, a następnie spróbować powtórzyć dany punkt instalacji, na co zwykle pozwala program instalacyjny (opcja: *Retry*). Opcję *Retry* warto zastosować nawet, jeśli nie jesteśmy w stanie niczego zmienić, aby dostosować się do uwag instalatora. Jeśli to nie pomoże, to należy zanotować komunikat instalatora i kontynuować instalację, na co

zwykle pozwala program instalacyjny po wyborze opcji *Ignore*. W wielu przypadkach tak przeprowadzona instalacja nie spowoduje istotnych dysfunkcyjności w działaniu SZBD Oracle. Oczywiście zależy to od rodzaju powstałego problemu. Dlatego zawsze musimy upewnić się w dokumentacji lub na forach użytkowników Oracle, jaka mogła być przyczyna kłopotów podczas instalacji.

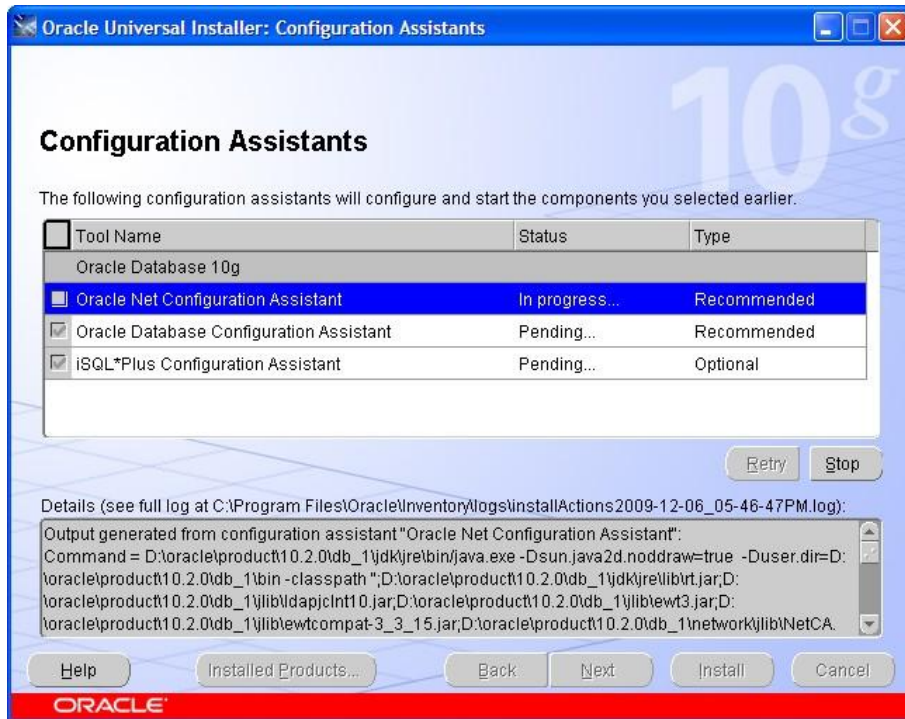


A.26 Po instalacji oprogramowania serwera program instalacyjny musi dokończyć jego konfigurację

Krok 9

Po instalacji SZBD Oracle 10g program instalacyjny ma za zadanie dokończyć konfigurację serwera (rys. A.26 - A.32). W tym celu uruchamia kolejno dwa lub trzy programy pomocnicze:

- Oracle Net Configuration Assistant – konfiguruje procesy nasłuchu sieciowego serwera bazy danych Oracle
- Oracle Database Configuration Assistant – tworzy bazę danych, jeśli ta opcja została wcześniej wybrana (patrz krok 7 opisu procedury instalacyjnej, w szczególności rys. A.18)
- iSQL*Plus Configuration Assistant – tworzy usługę iSQL*Plus wygodnego terminala SQL dostępnego poprzez przeglądarki internetowe



A.27 Konfiguracja procesów nasłuchu sieciowego serwera bazy danych, która kończy się pomyślnie przedstawionym poniżej komunikatem.

```
-----
Output generated from configuration assistant
"Oracle Net Configuration Assistant":
Command =
D:\oracle\product\10.2.0\db_1\jdk\jre\bin/java.
exe -Dsun.java2d.noddraw=true -
Duser.dir=D:\oracle\product\10.2.0\db_1\bin -
classpath
";D:\oracle\product\10.2.0\db_1\jdk\jre\lib\rt.
jar;D:\oracle\product\10.2.0\db_1\jlib\ldapjcln
t10.jar;D:\oracle\product\10.2.0\db_1\jlib\ewt3
.jar;D:\oracle\product\10.2.0\db_1\jlib\ewtcomp
at-
3_3_15.jar;D:\oracle\product\10.2.0\db_1\networ
k\jlib\NetCA.jar;D:\oracle\product\10.2.0\db_1\
network\jlib\netcam.jar;D:\oracle\product\10.2.
0\db_1\jlib\netcfg.jar;D:\oracle\product\10.2.0
\db_1\jlib\help4.jar;D:\oracle\product\10.2.0\d
b_1\jlib\jewt4.jar;D:\oracle\product\10.2.0\db_
1\jlib\oracle_ice.jar;D:\oracle\product\10.2.0\
db_1\jlib\share.jar;D:\oracle\product\10.2.0\db
_1\jlib\swingall-
1_1_1.jar;D:\oracle\product\10.2.0\db_1\jdk\jre
\lib\i18n.jar;D:\oracle\product\10.2.0\db_1\jli
b\srvmhas.jar;D:\oracle\product\10.2.0\db_1\jli
b\srvm.jar;D:\oracle\product\10.2.0\db_1\oui\jlib\

```

```
ib\OraInstaller.jar;D:\oracle\product\10.2.0\db_1\lib\xmlparserv2.jar;D:\oracle\product\10.2.0\
\db_1\network\tools" oracle.net.ca.NetCA
/orahome D:\oracle\product\10.2.0\db_1 /orahnam
OraDb10g_home1 /instype typical /inscomp
client,oraclenet,javavm,server,ano /insprtcl
tcp,nmp /cfg local /authadp NO_VALUE /nodeinfo
NO_VALUE /responseFile
D:\oracle\product\10.2.0\db_1\network\install\n
etca_typ.rsp
```

Parsing command line arguments:

```
Parameter "orahome" =
D:\oracle\product\10.2.0\db_1
Parameter "orahnam" = OraDb10g_home1
Parameter "instype" = typical
Parameter "inscomp" =
client,oraclenet,javavm,server,ano
Parameter "insprtcl" = tcp,nmp
Parameter "cfg" = local
Parameter "authadp" = NO_VALUE
Parameter "nodeinfo" = NO_VALUE
Parameter "responsefile" =
D:\oracle\product\10.2.0\db_1\network\install\n
etca_typ.rsp
```

Done parsing command line arguments.

Oracle Net Services Configuration:

Profile configuration complete.

Listener started at port:1521

Oracle Net Listener Startup:

Running Listener Control:

```
D:\oracle\product\10.2.0\db_1\bin\lsnrctl
start LISTENER
```

Listener Control complete.

Listener started successfully.

Setting Listener service to start automatically.

Listener configuration complete.

Default local naming configuration complete.

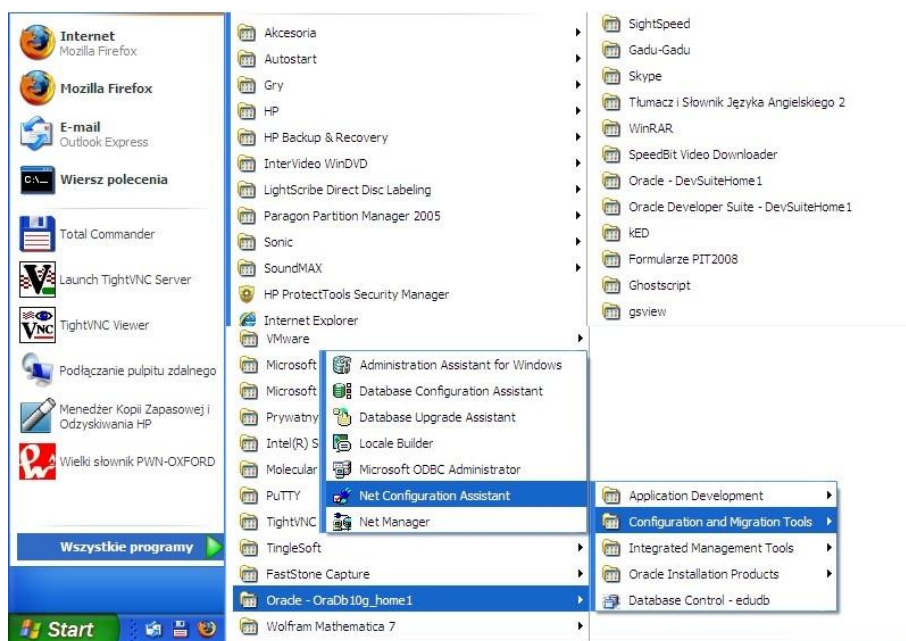
Oracle Net Services configuration successful.

The exit code is 0

Configuration assistant "Oracle Net
Configuration Assistant" **succeeded**

Poprawne ustawienie sieciowego nasłuchu serwera bazy danych Oracle jest niezwykle istotne, ponieważ pozwala skonfigurować autentykację sieciową dostępu do serwera bazy danych, która

jest najczęściej stosowaną formą weryfikacji szczególnie zdalnych użytkowników bazy danych. Więcej na temat form autentykacji połączeń z serwerem bazy danych Oracle można znaleźć w dalszej części opisu kroku 9 oraz w dodatkach C i E. Natomiast tutaj zostanie pokazane, co należy zrobić, gdy konfiguracja wykonywana przez Oracle Net Configuration Assistant nie powiedzie się, tzn. gdy nie uzyskamy statusu *Succeeded* dla zadania, które miał wykonać ten program. Wówczas powinniśmy powtórzyć próbę konfiguracji nasłuchu sieciowego serwera bazy danych, wybierając opcję *Retry*. Gdy to nie przyniesie pożądanego skutku, należy samodzielnie uruchomić Oracle Net Configuration Assistant w sposób pokazany na rys. A.28 (menu *Start* → *Oracle – OraDb10g_home1* → *Configuration and Migration Tools* → *Net Configuration Assistant*)



Rys. A.28 Uruchamianie Oracle Net Configuration Assistant poprzez menu *Start* systemu Windows

lub wykonując plik wsadowy `ORACLE_HOME\bin\netca.bat`, gdzie `ORACLE_HOME` oznacza katalog, w którym został zainstalowany SZBD Oracle. W naszym przykładzie (krok 5, rys. A.15):

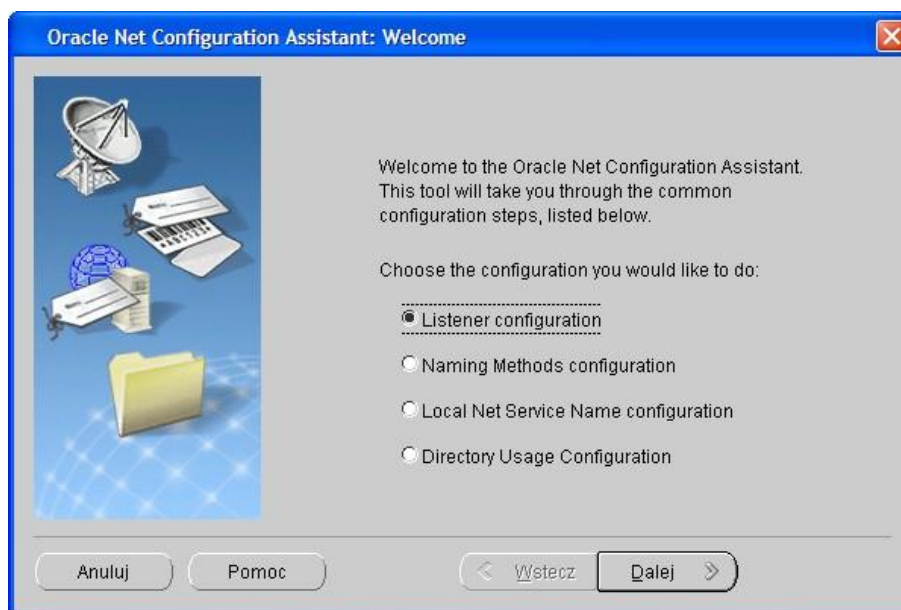
```
ORACLE_HOME = D:\oracle\product\10.2.0
```

Zatem Oracle Net Configuration Assistant dla naszej instalacji SZBD moglibyśmy uruchomić z konsoli systemu operacyjnego Windows, wpisując:

```
D:\oracle\product\10.2.0\bin\netca.bat
```

Korzystając z Oracle Net Configuration Assistant powinniśmy skonfigurować parametry sieciowego nasłuchu serwera bazy danych oraz szczegółowe metody autentykacji dostępu do serwera. Dla naszej przykładowej instalacji można to zrobić w sposób przedstawiony na rys. A.29 – A.38.

- Konfigurację parametrów nasłuchu sieciowego serwera bazy danych, które są zapisywane do pliku tekstowego `ORACLE_HOME\NETWORK\ADMIN\listener.ora` (w naszym przykładzie `D:\oracle\product\10.2.0\NETWORK\ADMIN\listener.ora`) można prześledzić na rys. A.29 - A.35
- Wybór szczegółowych metod autentykacji dostępu klienta do serwera bazy danych oraz kolejności ich stosowania, które są zapisywane do pliku tekstowego `ORACLE_HOME\NETWORK\ADMIN\sqlnet.ora` (w naszym przykładzie `D:\oracle\product\10.2.0\NETWORK\ADMIN\sqlnet.ora`) można prześledzić na rys. A.36 – A.38



Rys. A.29 Rozpoczynamy konfigurację parametrów nasłuchu sieciowego serwera bazy danych Oracle



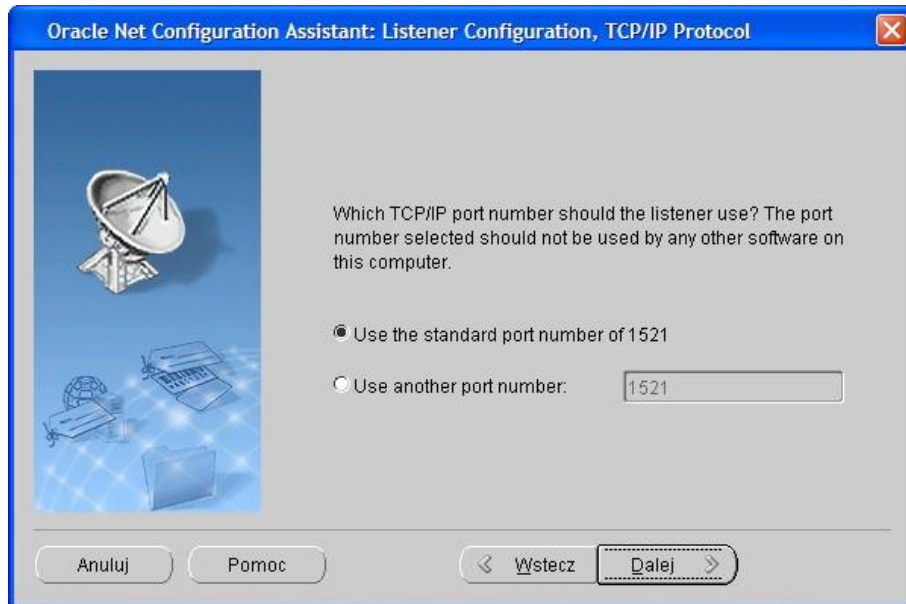
A.30 Dodajemy nowy proces sieciowego nasłuchu serwera bazy danych



A.31 Nadajemy nazwę procesowi nasłuchu sieciowego serwera bazy danych



- A. 32 Wybieramy protokoły połączeń akceptowane przez proces sieciowego nasłuchu serwera bazy danych



- A. 33 Ustalamy numer portu dla wybranego wcześniej protokołu TCP



A. 34 Ustalamy wartość klucza dla wybranego wcześniej protokołu IPC



A. 35 Kończymy konfigurację procesu nasłuchu sieciowego serwera bazy danych

W efekcie do pliku `listener.ora` trafiają następujące wpisy:

```
# listener.ora Network Configuration File:  
D:\oracle\product\10.2.0\db_1\network\admin\listener.ora  
# Generated by Oracle configuration tools.
```

```

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = D:\oracle\product\10.2.0\db_1)
      (PROGRAM = extproc)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1))
      (ADDRESS = (PROTOCOL = TCP) (HOST = BMW.) (PORT = 1521))
    )
  )

```

Oznaczają one, że utworzyliśmy proces nasłuchu sieciowego o nazwie LISTENER związany z instalacją SZBD Oracle, która znajduje się w katalogu (ORACLE_HOME) D:\oracle\product\10.2.0\db_1. Proces LISTENER będzie akceptował połączenia, wykorzystujące protokół TCP na porcie 1521 oraz protokół IPC o wartości klucza EXTPROC1. O ile powszechnie znany TCP (*Transmission Control Protocol*) nie wymaga wyjaśnień, to należy zwrócić uwagę, że protokół IPC (*Inter-Process Communication*) nie jest protokołem sieciowym i jest używany do obsługi lokalnych połączeń, tzn. gdy klient i serwer bazy danych są na jednym komputerze. Warto przy tym zaznaczyć, że w tej sytuacji lokalny klient bazy danych może komunikować się z serwerem bazy danych także za pomocą protokołu TCP. Najczęściej jest to możliwe dzięki odpowiednim ustawieniom, które są typowe dla sieciowych klientów bazy danych, tzn. dzięki konfiguracji lokalnych usług nazw sieciowych, która będzie omówiona w dodatku C.

Następnie wybieramy metody autentykacji dostępu klienta do serwera bazy danych (rys. A.36 – A.38).



A. 36 Rozpoczynamy konfigurację szczegółowych metod autentykacji dostępu klienta do serwera bazy danych



A. 37 Wybieramy szczegółowe metod autentykacji dostępu klienta do serwera bazy danych



A.38 Kończymy konfigurację szczegółowych metod autentykacji dostępu klienta do serwera bazy danych

W efekcie do pliku `sqlnet.ora` trafiają następujące wpisy:

```
# sqlnet.ora Network Configuration File:
D:\oracle\product\10.2.0\db_1\network\admin\sqlnet.ora
# Generated by Oracle configuration tools.

# This file is actually generated by netca. But if customers choose to
# install "Software Only", this file won't exist and without the native
# authentication, they will not be able to connect to the database on NT.

SQLNET.AUTHENTICATION_SERVICES= (NTS)

NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
```

Oznaczają one, że nasz serwer bazy danych dopuszcza:

- Autentykację przez system operacyjny Windows (parametr `NTS`), dzięki której użytkownicy systemu operacyjnego należący do grupy administratorów bazy danych Oracle (grupa

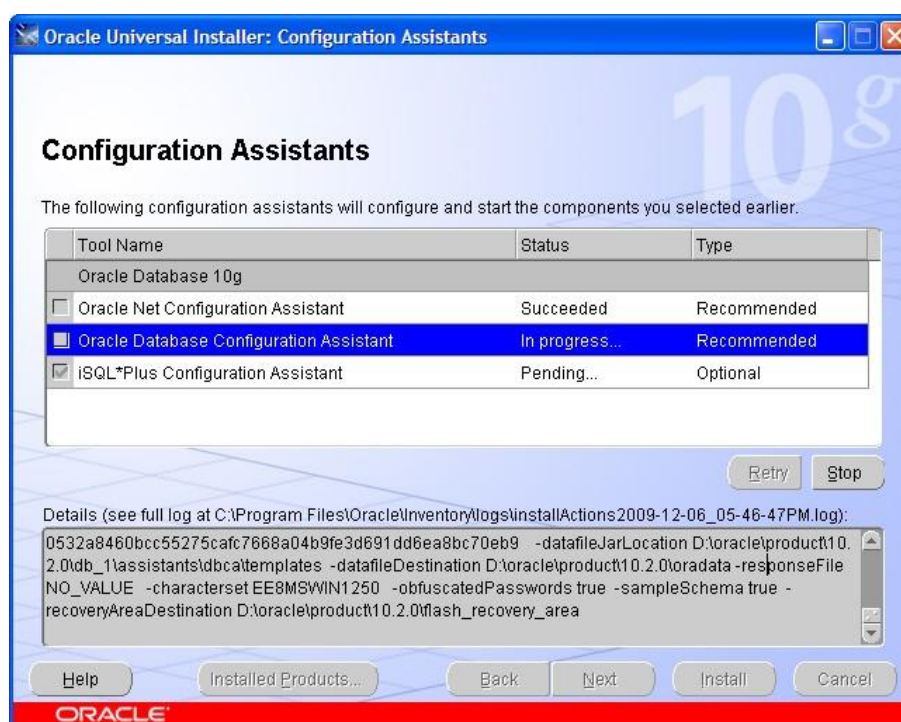
ORA_DBA) będą mogli logować się do bazy danych jak użytkownik bazy danych o najwyższych uprawnieniach administracyjnych jedynie na podstawie swojego logowania do systemu operacyjnego, tzn. bez konieczności podania zarówno nazwy odpowiedniego administracyjnego użytkownika bazy danych, jak i jego hasła.

- Typową, powszechnie stosowaną formę autentykacji sieciową dostępu klientów do bazy danych, wykorzystującą lokalną (tzn. skonfigurowaną na komputerze klienta) usługę sieciową nazwę bazy danych (parametr TNSNAMES).
- Uproszczoną formę autentykacji sieciowej dostępu klientów do bazy danych, która nie wymaga zdefiniowania sieciowej nazwy bazy danych na komputerze klienta (parametr EZCONNECT).

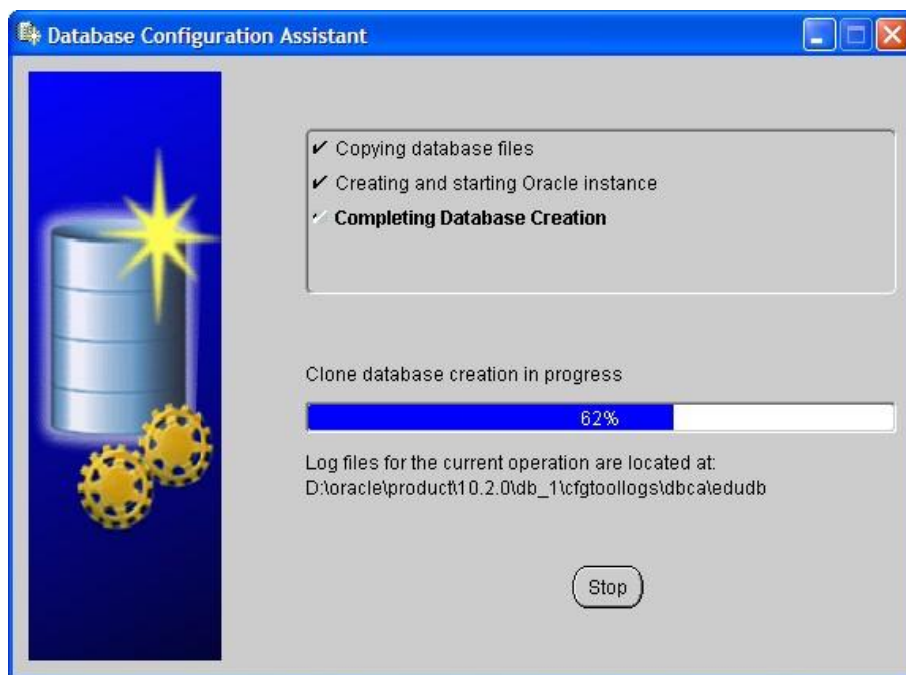
Kolejność występowania parametrów w NAMES.DIRECTORY_PATH decyduje o kolejności wyboru metody autentykacji przez serwer bazy danych.

Więcej o metodach autentykacji z punktu widzenia klienta bazy danych można znaleźć w dodatkach C i E.

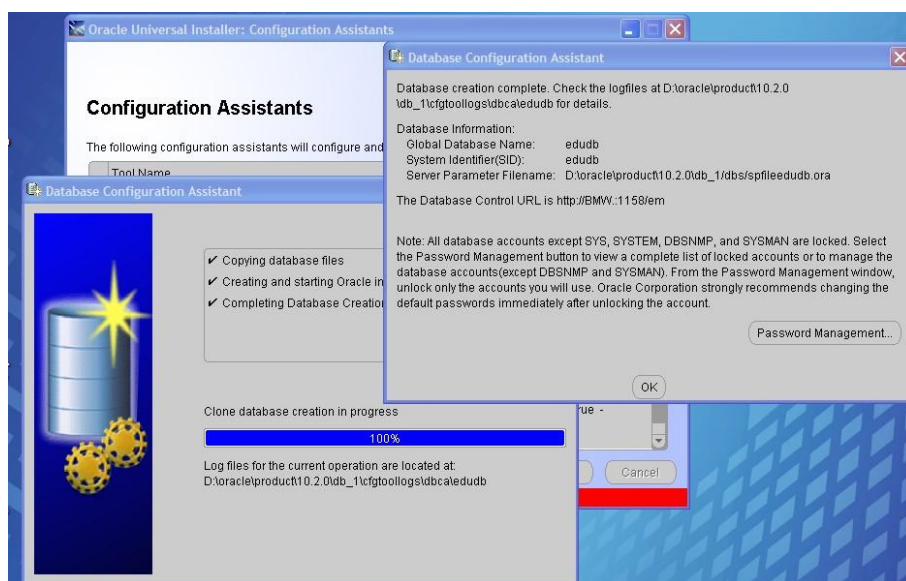
Po prawidłowo zakończonej konfiguracji nasłuchu sieciowego serwera tworzona jest baza danych (rys. A.39 - A.42) zgodnie z wybranymi parametrami w kroku 7 (rys. A.18 - A.24).



A.39 Tworzenie bazy danych



A.40 Końcowy etap tworzenia bazy danych



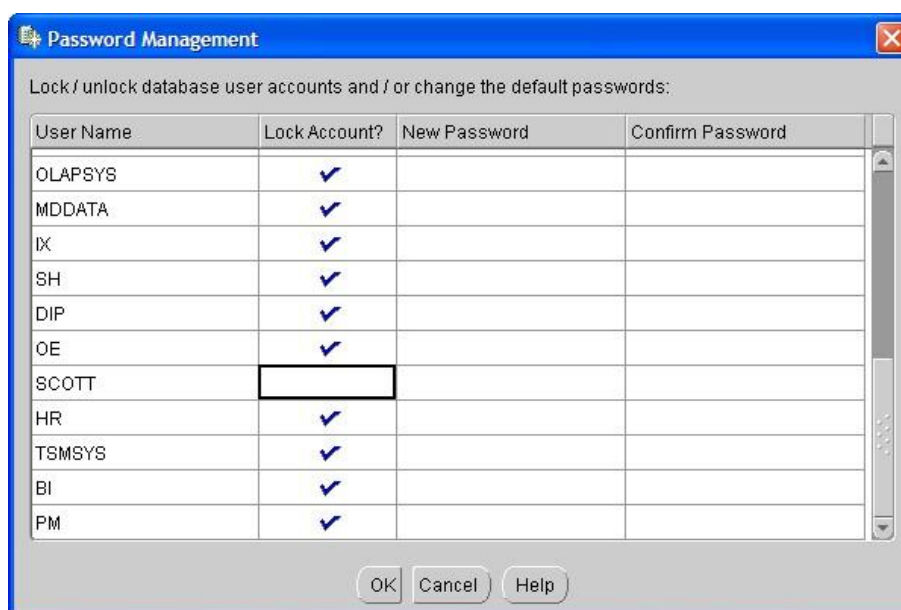
A.41 Po zakończeniu konfiguracji serwera bazy danych i tworzenia bazy danych program instalacyjny daje możliwość zarządzania kontami użytkowników bazy (opcja *Password Management*)

Po zakończeniu procesu tworzenia bazy danych otrzymujemy komunikat o bardzo ważnych parametrach bazy danych (rys. A.41), które bezwzględnie należy zapamiętać, takich jak globalna nazwa bazy danych (*Global Database Name*), identyfikator systemu (instancji) bazy danych (SID – *System Identifier*) oraz położenie i nazwa pliku parametrów

(inicjalizacyjnych) serwera bazy danych (*Server Parameter Filename*), którego nazwa posiadająca prefiks *spfile* oznacza, że jest to nowsza wersja (dynamiczna) pliku parametrów inicjalizacyjnych serwera bazy danych Oracle (nazwa statycznego pliku parametrów inicjalizacyjnych charakteryzowałaby się słowem *pfile*). Dynamiczny plik parametrów daje możliwość zmieniania niektórych parametrów serwera bazy danych w trakcie jego pracy bez konieczności restartu bazy danych.

Po utworzeniu bazy danych mamy również możliwość wstępnego zarządzania kontami użytkowników utworzonej bazy, dzięki udostępnionej opcja *Password Management* na rys. A.41. Domyślnie odblokowane są tylko konieczne schematy użytkowników administracyjnych (np. SYS i SYSTEM). Natomiast schematy innych użytkowników są zablokowane, w tym również edukacyjny schemat użytkownika SCOTT, który został utworzony dzięki zaznaczeniu opcji *Create database with sample schemas* (rys. A.20), gdy ustalaliśmy parametry bazy danych. Jeśli chcemy mieć dostęp do schematu użytkownika SCOTT o domyślnym hasle TIGER (wielkość znaków nazwach użytkowników i hasłach nie ma znaczenia do wersji do wersji bazy Oracle 10g), to należy odznaczyć zablokowanie konta SCOTT w sposób pokazany na rys. A.42. Jeśli zapomnimy odblokować konto jakiegoś użytkownika w końcowym etapie instalacji, to możemy to zrobić później. W tym celu logujemy się do bazy danych na konto użytkownika administracyjnego (np. SYS lub SYSTEM) i chcąc np. odblokować konto SCOTT wykonujemy komendę:

```
alter user scott account unlock;
```



A.42 Ręczne odblokowanie konta edukacyjnego użytkownika SCOTT

Możliwość wstępnego zarządzania użytkownikami kończy tworzenie i konfigurację bazy danych i instalator wyświetla krótki komunikat o tym, w jaki sposób była tworzona baza i czy ta procedura została prawidłowo zakończona. W przypadku naszej przykładowej bazy danych taki komunikat był następujący:

```
-----  
Output generated from configuration assistant  
"Oracle Database Configuration Assistant":  
Command = C:\WINDOWS\system32\cmd /c call  
D:\oracle\product\10.2.0\db_1\bin\dbca.bat -  
progress_only -createDatabase -templateName  
General_Purpose.dbc -gdbName edudb -sid edudb  
-sysPassword  
0530408500d9e13c89aeac41efa034f38ef89a36e24d703  
3ac -systemPassword  
05fe2ee300da27483b53ff52c382f2dc6595c05efdc9914  
f7a -sysmanPassword  
05fe2ee300da27483bda7e69cbe0251c03dad0fca3d3606  
7c7 -dbSNMPPassword  
05fe2ee300da27483b74241e87a50c8ffe39dd0be53cf11  
3bf -emConfiguration NOEMAIL -hostUserName  
Administrator -hostUserPassword  
0532a8460bcc55275cafc7668a04b9fe3d691dd6ea8bc70  
eb9 -datafileJarLocation  
D:\oracle\product\10.2.0\db_1\assistants\dbca\t  
emplates -datafileDestination  
D:\oracle\product\10.2.0\oradata -responseFile  
NO_VALUE -characterset EE8MSWIN1250 -  
obfuscatedPasswords true -sampleSchema true -  
recoveryAreaDestination  
D:\oracle\product\10.2.0\flash_recovery_area  
  
Configuration assistant "Oracle Database  
Configuration Assistant" succeeded  
-----
```

Na zakończenie tworzona jest jeszcze usługa isqlplus, której uruchomienie na serwerze pozwala używać wygodny terminal SQL dostępny zdalnie i lokalnie poprzez przeglądarkę internetową. Także po tej operacji program instalacji wydaje komunikat i tak w przypadku naszej przykładowej instalacji otrzymujemy:

```
-----  
Output generated from configuration assistant  
"iSQL*Plus Configuration Assistant":  
Command =  
D:\oracle\product\10.2.0\db_1\bin\isqlplusctl.b  
at start  
iSQL*Plus 10.2.0.1.0
```

```
Copyright (c) 2003, 2005, Oracle. All rights reserved.
Starting iSQL*Plus ...
iSQL*Plus started.
```

```
Configuration assistant "iSQL*Plus
Configuration Assistant" succeeded
```

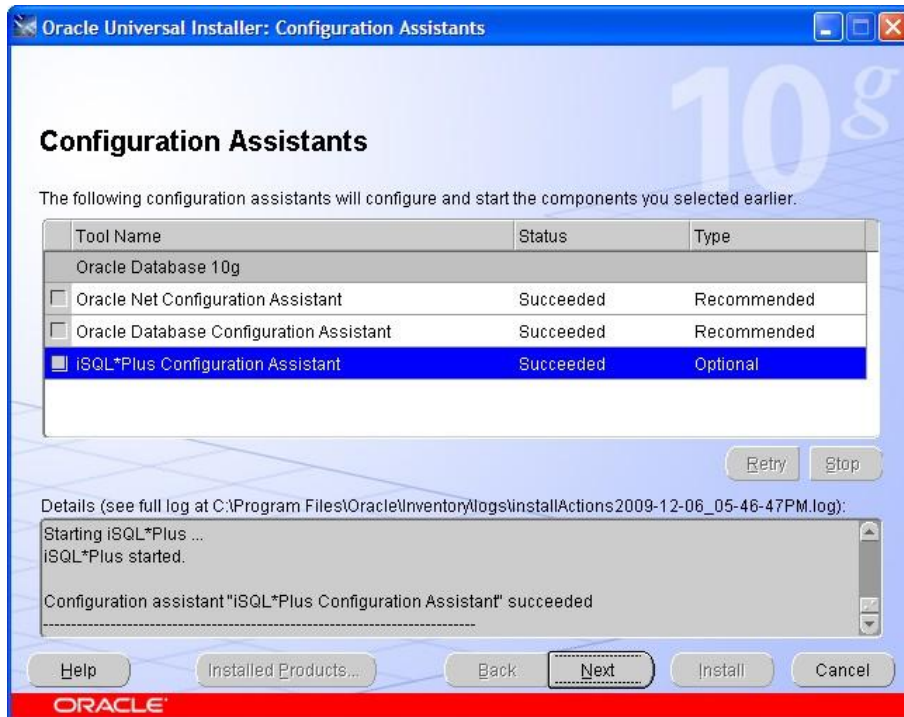
Ponadto otrzymujemy wiadomość, z której dowiadujemy się, gdzie zostały zapisane skrypty zawierające wszystkie komendy, które można wykonać poza graficznym interfejsem użytkownika (OUI – *Oracle User Interface*), aby uruchomić bez pomocy OUI wszystkie programy omawiane w tym kroku instalacji. W przypadku naszej przykładowej instalacji jest to skrypt

```
D:\oracle\product\10.2.0\db_1\cfgtoollogs/configToolAllCommands
```

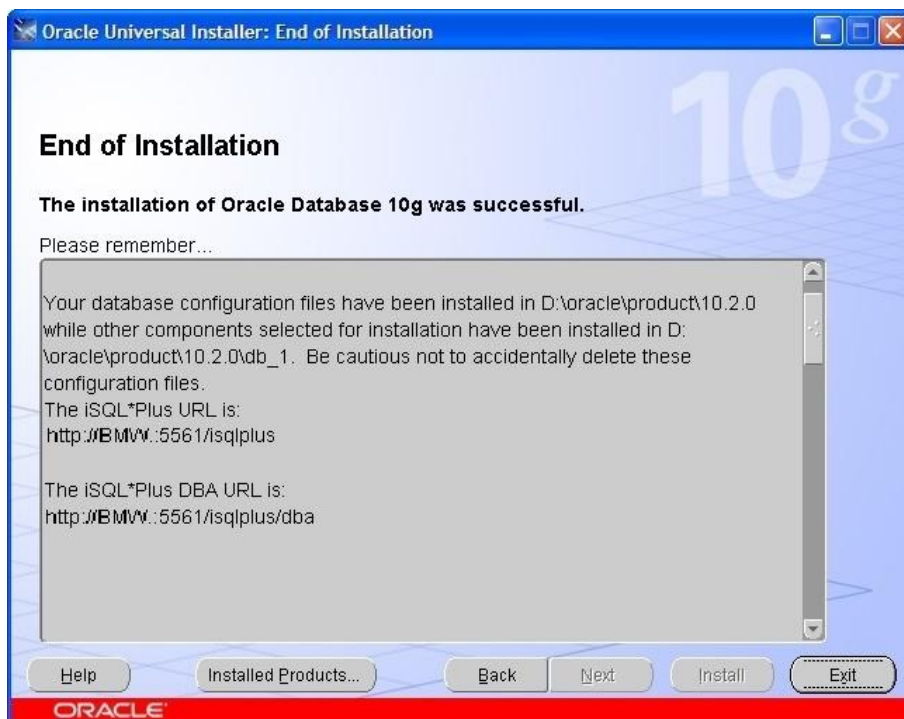
zgodnie z poniższym komunikatem

```
-----
The
"D:\oracle\product\10.2.0\db_1\cfgtoollogs/conf
igToolAllCommands" script contains all commands
to be executed by the configuration assistants.
This file may be used to run the configuration
assistants outside of OUI. Note that you may
have to update this script with passwords (if
any) before executing the same.
-----
```

Na tym instalacja SZDB Oracle wraz z tworzeniem bazy danych zostaje zakończona (rys. A.43). Na koniec dostajemy jeszcze kilka wskazówek (rys. A.44), np. o tym jak uruchomić isqlplus, który w naszym przykładzie powinien być dostępny przez przeglądarkę internetową pod następującym adresem URL: <http://BMW.:5561/isqlplus>. Warto zwrócić uwagę na numer portu 5561, na którym jest udostępniana usługa isqlplus. Jest on inny niż typowo ustalany 5560, co wskazuje na to, że na tym komputerze była już wcześniej wykonywana próba zainstalowania usługi isqlplus, np. dla innej instancji bazy danych Oracle 10g. Jeśli była ona udana, to na porcie 5560 może być aktywna usługa isqlplus dla innej instancji bazy danych, jeśli zostanie ona prawidłowo uruchomiona.



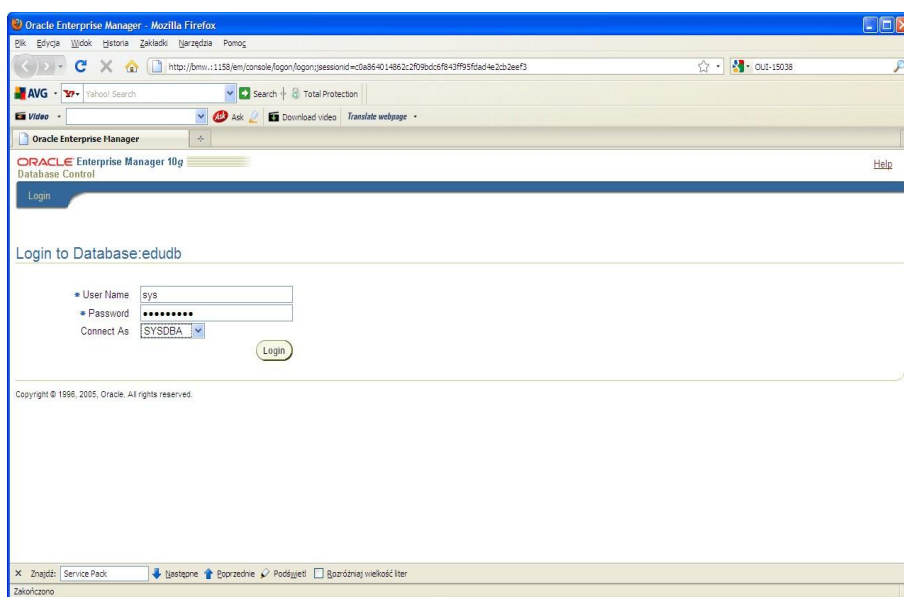
A.43 Konfiguracja serwera bazy danych i tworzenie bazy danych zostały zakończone



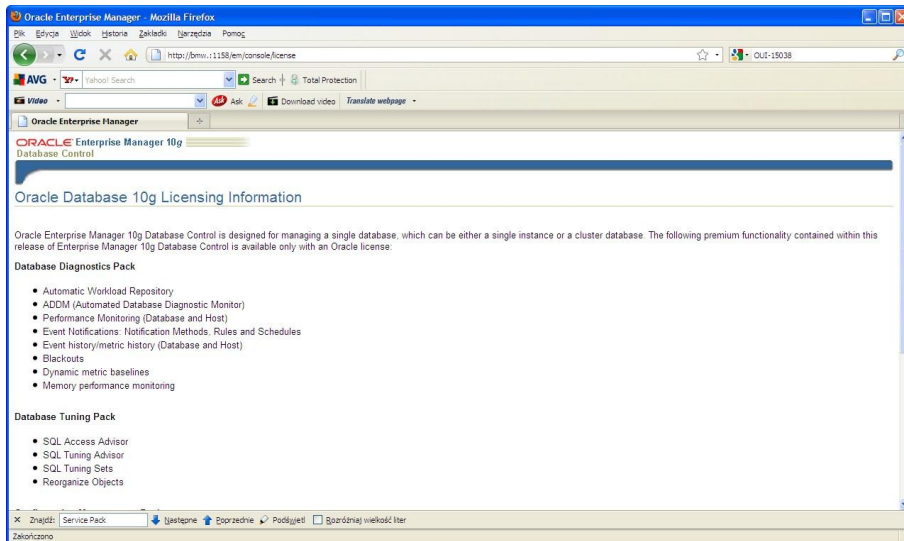
A.44 Koniec instalacji

Po prawidłowo przeprowadzonej instalacji SZBD Oracle wraz z tworzeniem bazy danych powinien uruchomić się *Oracle Enterprise Manager*, który jest wygodnym, dostępnym poprzez przeglądarkę internetową narzędziem, ułatwiającym administrowanie bazą danych. Jak widać na rys. A.45, zalogować się do bazy z poziomu tego narzędzia mogą wyłącznie użytkownicy o najwyższych uprawnieniach (standardowo jest to użytkownik SYS) i wyłącznie w specjalnym trybie (najlepiej w trybie SYSDBA). Podobnie jak *isqlplus* dostęp do usługi *Enterprise Manager* (em), jeśli została ona uruchomiona na serwerze, jest możliwy zarówno zdalnie, jak i lokalnie pod odpowiednim adresem URL. W naszym przykładzie ten adres (<http://BMW.:1158/em>) jest nietypowy, ponieważ nazwa domenowa komputera, na którym została utworzona baza danych i został uruchomiony *Oracle Enterprise Manager* ma pustą domenę. Pomimo tego program instalacyjny był w stanie poinformować nas po zakończeniu instalacji (rys. A.41) o tym, jaki jest adres URL (*The Database Control URL*) usługi *Oracle Enterprise Manager*.

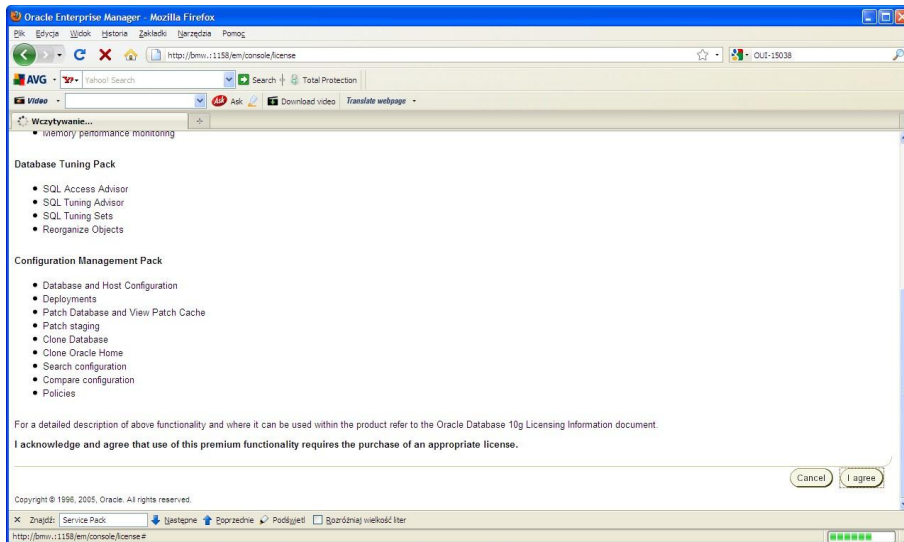
Pierwsze zalogowanie do *Oracle Enterprise Manager* skutkuje wyświetleniem informacji o warunkach licencji (rys. A.46 - A.47). Lektura tego tekstu pokazuje, że należy sprawdzić w odpowiednich dokumentach otrzymanych po zakupie, czy posiadamy odpowiednią licencję, ponieważ *Oracle Enterprise Manager*, jest często narzędziem dodatkowo płatnym. Przyjęcie do wiadomości tej informacji, potwierdzone wyborem opcji *I agree* powoduje otwarcie strony głównej (A.48) narzędzia *Oracle Enterprise Manager* naszej bazy *edudb*.



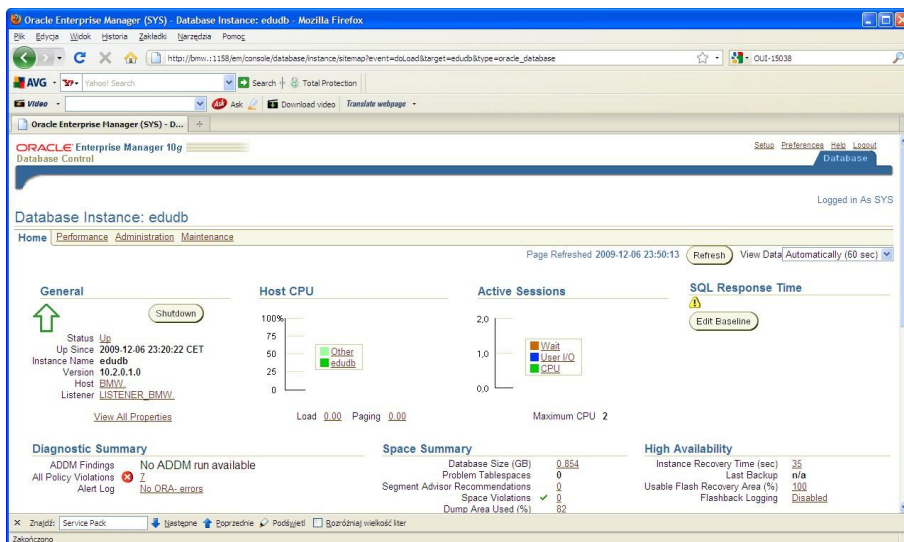
A.45 Pierwsze uruchomienie narzędzia *Oracle Enterprise Manager* w celu zaakceptowania informacji o warunkach licencji



A.46 Początek informacji o warunkach licencji



A.47 Koniec informacji o warunkach licencji z możliwością ich odrzucenia (*Cancel*) bądź zaakceptowania (*I agree*)

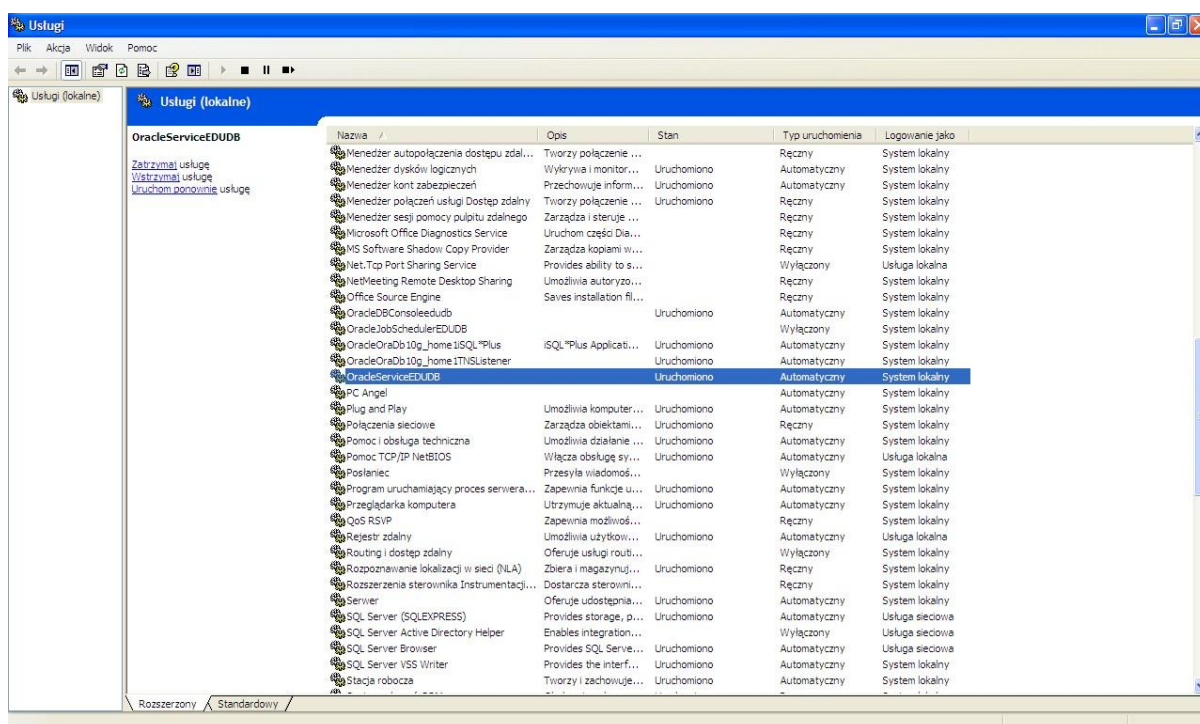


A.48 Strona główna Oracle Enterprise Manager naszej bazy edudb

Opis możliwości *Oracle Enterprise Manager* wykracza poza ramy tego skryptu, dlatego wszystkich zainteresowanych odsyłamy do oficjalnej dokumentacji, dostępnej np. pod adresem <http://www.oracle.com/technetwork/documentation/oem102-091877.html>.

B. Obsługa podstawowych usług serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional

Z uruchomioną bazą danych Oracle w środowisku Windows jest związanych kilka usług systemu operacyjnego. Wygodnym narzędziem obsługującym usługi systemu operacyjnego Windows XP jest menadżer usług dostępny w *Narzędziach administracyjnych* systemu Windows: *Start* → *Panel sterowania* → *Wydajność i konserwacja* → *Narzędzia administracyjne* → *Usługi*. Na rys. B.1 został pokazany menadżer usług, dzięki któremu możemy zobaczyć usługi systemu operacyjnego Windows związane z bazą danych Oracle 10g o nazwie *edudb*, której instalacja została opisana w dodatku A.



B.1 Usługi systemu operacyjnego Windows, związane z bazą danych Oracle 10g o nazwie *edudb*, widziane za pomocą menadżera usług systemu Windows

Skupimy się tutaj na czterech podstawowych usługach spośród pięciu, które są związane z bazą danych Oracle 10g w systemie Windows XP Professional:

`OracleOraDb10g_home1TNSListener` – usługa procesu nasłuchu sieciowego serwera bazy danych Oracle 10g, którą możemy uruchamiać i zatrzymywać w menadżerze usług systemu operacyjnego Windows. Warto pamiętać, że po prawidłowo przeprowadzonej instalacji systemu zarządzania bazą danych to samo, jak też zostało to opisane w dodatku E, możemy wykonać z konsoli systemu operacyjnego Windows za pomocą komend:

`lsnrctl start` – uruchomienie procesu nasłuchu sieciowego serwera bazy danych Oracle

`lsnrctl stop` – zatrzymanie procesu nasłuchu sieciowego serwera bazy danych Oracle

`OracleServiceEDUDB` – usługa instancji bazy danych `edudb`. Typowo, po utworzeniu bazy danych uruchomienie tej usługi zakończone jest otwarciem bazy danych. Jednak możemy tę usługę skonfigurować inaczej tzn. tak, aby po włączeniu usługi nie była otwierana baza danych. Jednak należy pamiętać, że usługa instancji bazy danych musi być uruchomiona, abyśmy byli w stanie połączyć z nieaktywną instancją (*idle instance*) bazy danych, tzn. z instancją bazy danych, która nie jest otwarta. Jak zostało to opisane w dodatku E, takie połączeniu w celu otwarcia bazy danych można nawiązać jako użytkownik systemu operacyjnego, który posiada uprawnienia administratora bazy danych, wykonując w konsoli systemu operacyjnego komendę:

```
sqlplus / as sysdba
```

A następnie w środowisku programu `sqlplus`, wydając komendę: `startup`

`OracleDBConsoleedudb` – usługa zdalnej konsoli administracyjnej bazy danych, czyli programu *Oracle Enterprise Manager*, będącego aplikacją przeglądarki internetowej, o którym można przeczytać w końcowej części dodatku A. Jak zostało to opisane w dodatku E, tę usługę można również uruchamiać i zatrzymywać z konsoli system operacyjnego za pomocą komend:

`emctl start dbconsole` – uruchomienie konsoli *Oracle Enterprise Manager* bazy danych

`emctl stop dbconsole` – zatrzymanie konsoli *Oracle Enterprise Manager* bazy danych

OracleOraDb10g_home1iSQL*Plus – usługa wygodnego terminala bazy danych *iSQL*Plus*, będącego aplikacją przeglądarki internetowej, o którym można przeczytać w końcowej części dodatku A. Tę usługę można również uruchamiać i zatrzymywać z konsoli system operacyjny za pomocą komend:

```
isqlplusctl start – uruchomienie iSQL*Plus
```

```
isqlplusctl stop – zatrzymanie iSQL*Plus
```

Należy podkreślić, że do prawidłowego, w pełni funkcjonalnego działania bazy danych najpierw należy uruchomić usługę procesu nasłuchu sieciowe, później usługę instancji bazy danych, a następnie kolejne usługi, np. dla *Oracle Enterprise Managera* i *iSQL*Plusa*, jeśli chcemy używać tego oprogramowania. Natomiast kolejność zatrzymywania tych usług powinna być odwrotna. Ponadto warto zauważyć, że w systemie Windows XP mamy jeszcze jedną metodę uruchamiania i zatrzymywania usług, wykorzystującą komendę `sc` systemu operacyjnego. Poniżej przedstawiono możliwą zawartość plików wsadowych `start.bat` i `stop.bat`, które w odpowiedniej kolejności uruchamiają i zatrzymują opisane usługi serwera bazy danych Oracle 10g.

Zawartość skryptu `start.bat`:

```
sc start OracleOraDb10g_home1TNSListener
```

```
sc start OracleServiceEDUDB
```

```
sc start OracleDBConsoleedudb
```

```
sc start OracleOraDb10g_home1iSQL*Plus
```

Zawartość skryptu `stop.bat`:

```
sc start OracleOraDb10g_home1iSQL*Plus
```

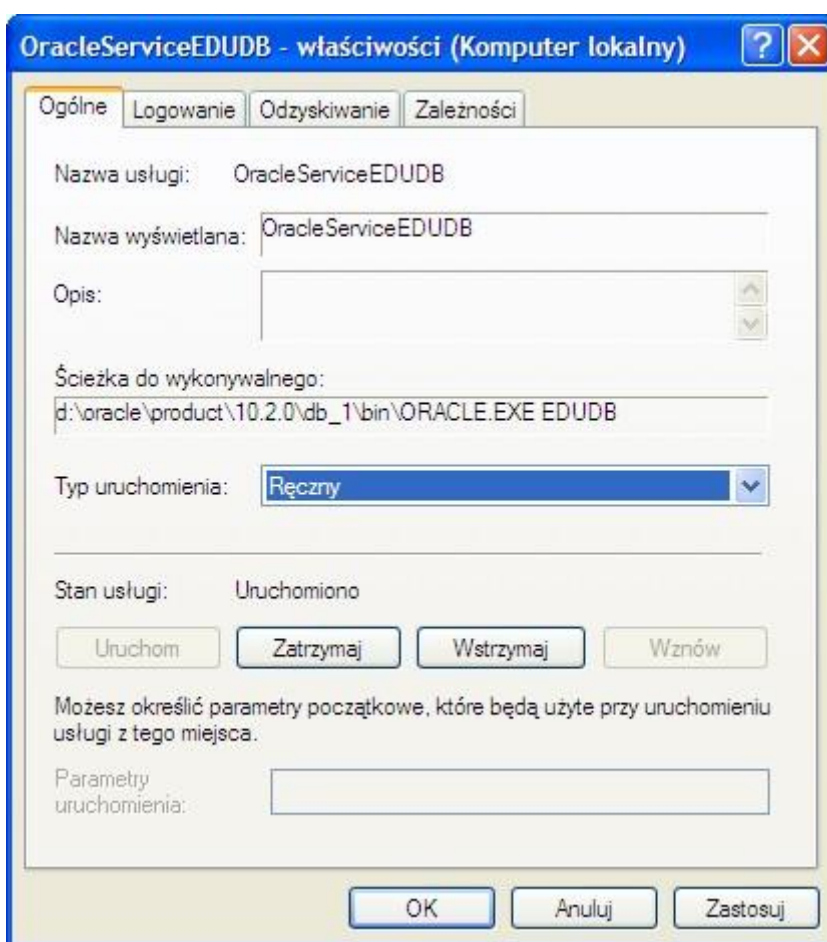
```
sc start OracleDBConsoleedudb
```



```
sc start OracleServiceEDUDB
```

```
sc start OracleOraDb10g_home1TNSListener
```

Wymienione powyżej cztery usługi serwera bazy danych Oracle 10g są po instalacji systemu zarządzania i utworzeniu bazy danych domyślnie ustawione w trybie automatycznego uruchamiania podczas uruchamiania systemu operacyjnego. Tryb *automatyczny* można zmienić w menadżerze usług na *ręczny*, jak pokazano na rys. B.2 w przypadku usługi instancji bazy danej edudb, a w razie potrzeby na *wyłączony*.



B.2 Zmiana trybu uruchamiania usługi instancji bazy danych edudb na ręczny we właściwościach tej usługi w menadżerze usług systemu Windows XP

Zmiany trybu uruchamiania usług możemy również wykonać za pomocą komendy `sc` systemu operacyjnego. Poniżej przedstawiono możliwą zawartość plików wsadowych `demand.bat` i `auto.bat`, które w odpowiedniej kolejności uruchamiają i zatrzymują opisane usługi serwera bazy danych Oracle 10g.

Zawartość skryptu `demand.bat`:

```
sc config OracleOraDb10g_home1TNSListener start= demand
```

```
sc config OracleServiceEDUDB start= demand
```

```
sc config OracleDBConsoleeedudb start= demand
```

```
sc config OracleOraDb10g_home1iSQL*Plus start= demand
```

Zawartość skryptu `auto.bat`:

```
sc config OracleOraDb10g_home1TNSListener start= auto
```

```
sc config OracleServiceEDUDB start= auto
```

```
sc config OracleDBConsoleeedudb start= auto
```

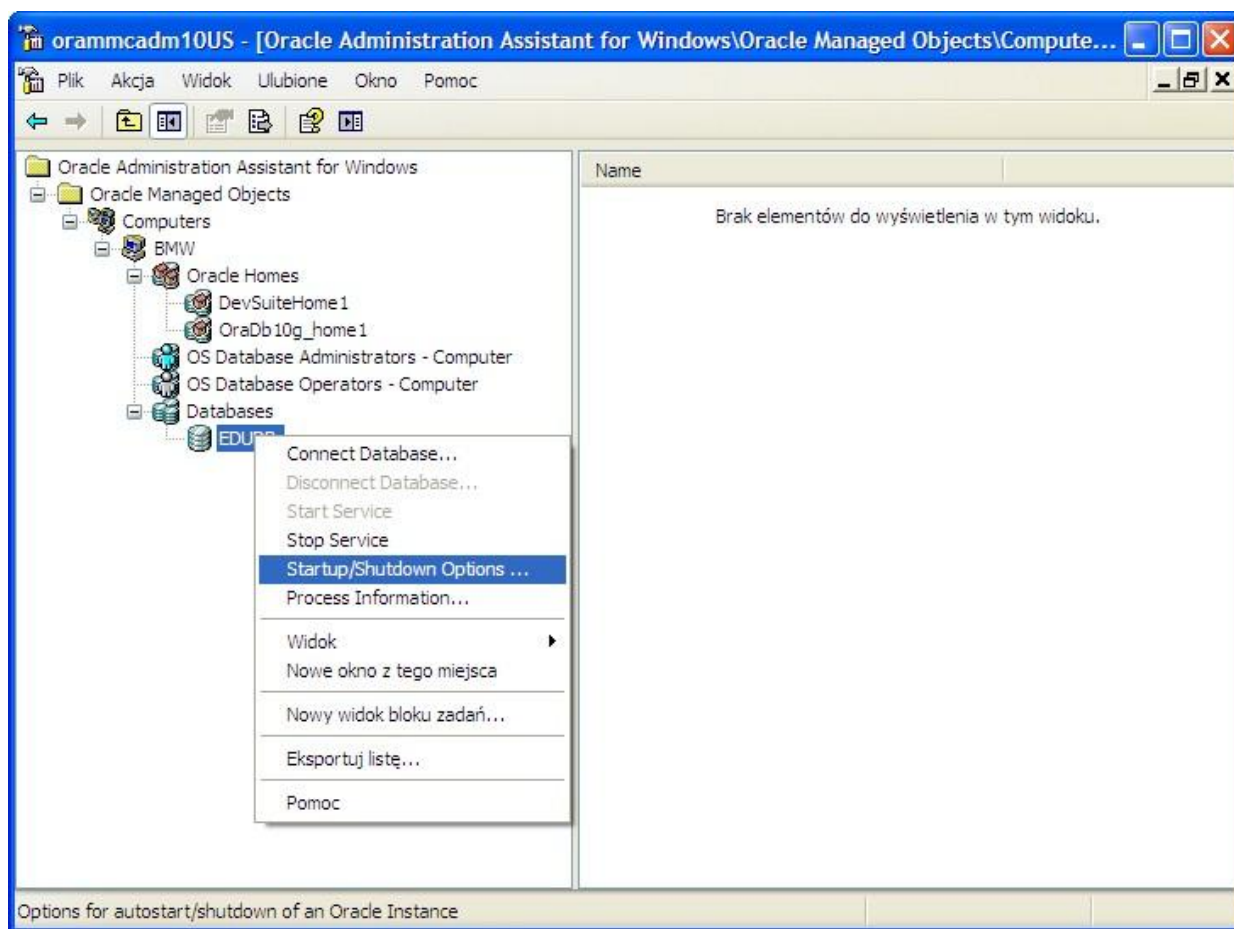
```
sc config OracleOraDb10g_home1iSQL*Plus start= auto
```

Rozpatrzmy jeszcze problem automatycznego otwierania bazy danych zaraz po uruchomieniu usługi instancji tej bazy danych. Jak już zostało wspomniane, taka jest domyślna konfiguracja serwera bazy danych. W systemie Windows możemy ją zmienić za pomocą narzędzia *Oracle Administration Assistant for Windows*, które znajdziemy poprzez menu *Start* systemu Windows. Przykładowo dla naszej instalacji opisanej w dodatku A ten program wspomagający administrację serwera bazy danych można uruchomić z poziomu:

Start → *Wszystkie programy* → *Oracle – OraDb10g_home1* → *Configuration and Migration Tools* → *Administration Assistant for Windows*. To narzędzie pozwala wykonać także inne

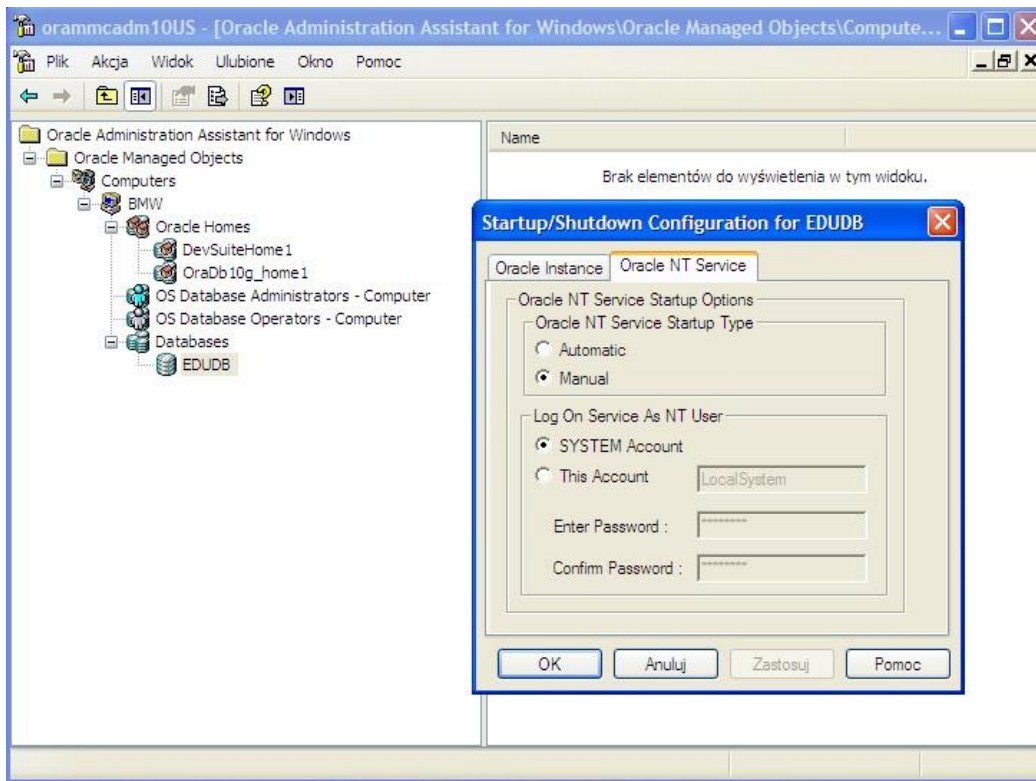
zmiany w konfiguracji serwera, m. in. udostępnia kolejny sposób określenia trybu uruchamiania usługi instancji bazy danych.

Na rysunkach B.3 – B.5 zostały pokazane naniesione zmiany w domyślnej konfiguracji serwera bazy danych obsługującego instancję bazy danych edudb. Po uruchomieniu *Oracle Administration Assistant for Windows* i odszukaniu naszej bazy danych w *Databases* klikamy prawym przyciskiem myszy na ikonce naszej bazy edudb i wybieramy opcję *Startup/Shutdown Options* (rys. B.3).

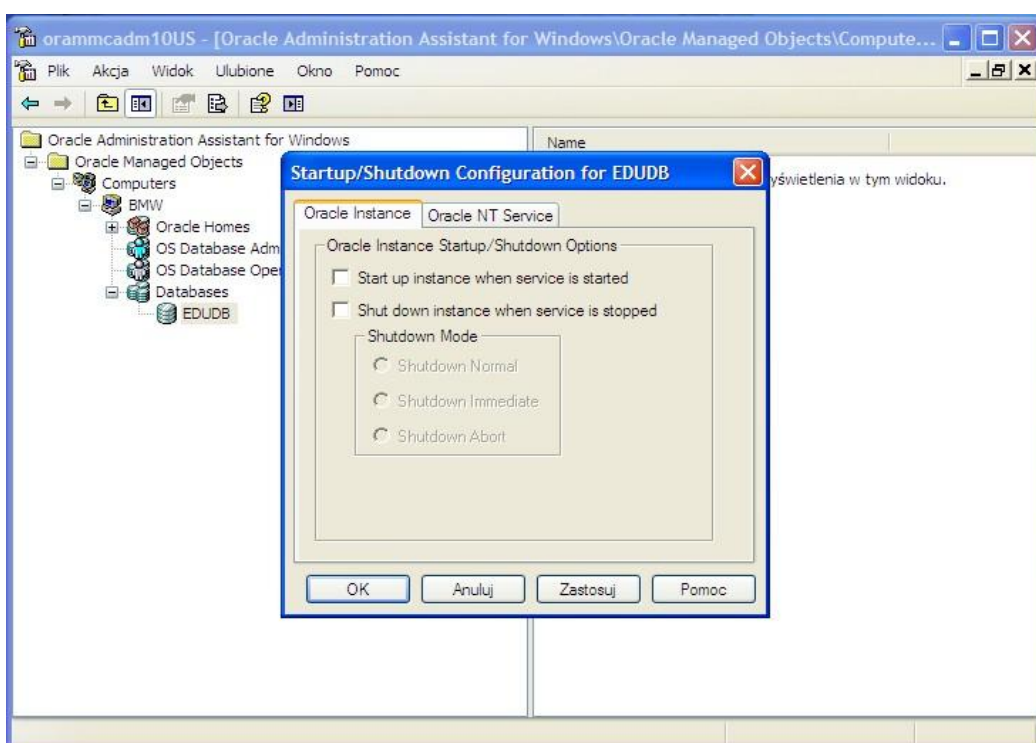


B.3 Wybór opcji *Startup/Shutdown Options* dla bazy danych edudb w narzędziu *Oracle Administration Assistant for Windows*

Następnie (rys. B. 4) w zakładce *Oracle NT Service* ustawiamy ręczne (opcja *Manual*) uruchamianie usługi instancji bazy danych edudb (czyli usługi o nazwie *OracleServiceEDUDB*) i klikamy *Zastosuj*. Natomiast w zakładce *Oracle Instance* (rys. B.5) odznaczamy opcję *Start up instance when service is started* i klikamy *Zastosuj*.



B.4 Konfiguracja ręcznego (opcja *Manual*) uruchamiania usługi instancji bazy danych edudb w narzędziu *Oracle Administration Assistant for Windows*



B.5 Konfiguracja bazy danych edudb w narzędziu *Oracle Administration Assistant for Windows*, która nie pozwala (odznaczona opcja *Start up instance when service is started*) na automatyczne otwarcie bazy edudb po uruchomieniu usługi instancji tej bazy.

Jak widać na rys. B.3 – B.5, w narzędziu *Oracle Administration Assistant for Windows* można wykonywać również inne zmiany w konfiguracji serwera bazy danych Oracle. Przykładowo w pozycjach *OS Database Administrators - Computer* oraz *OS Database Operators - Computer* można wskazać użytkowników systemu operacyjnego, którzy odpowiednio otrzymują uprawnienia administratora bazy danych oraz operatora bazy danych. Jeśli dany użytkownik systemu operacyjnego powinien mieć możliwość otwierania i zamykania bazy danych po uzyskaniu dostępu administracyjnego do serwera bazy na podstawie autentykacji użytkownika poprzez system operacyjny, opisanej szczegółowo w dodatku E, to należy go dołączyć do grupy administratorów bazy danych w pozycji *OS Database Administrators – Computer*. Domyślnie takim administratorem bazy danych jest użytkownik systemu operacyjnego, który przeprowadził instalację systemu zarządzania bazą danych Oracle.

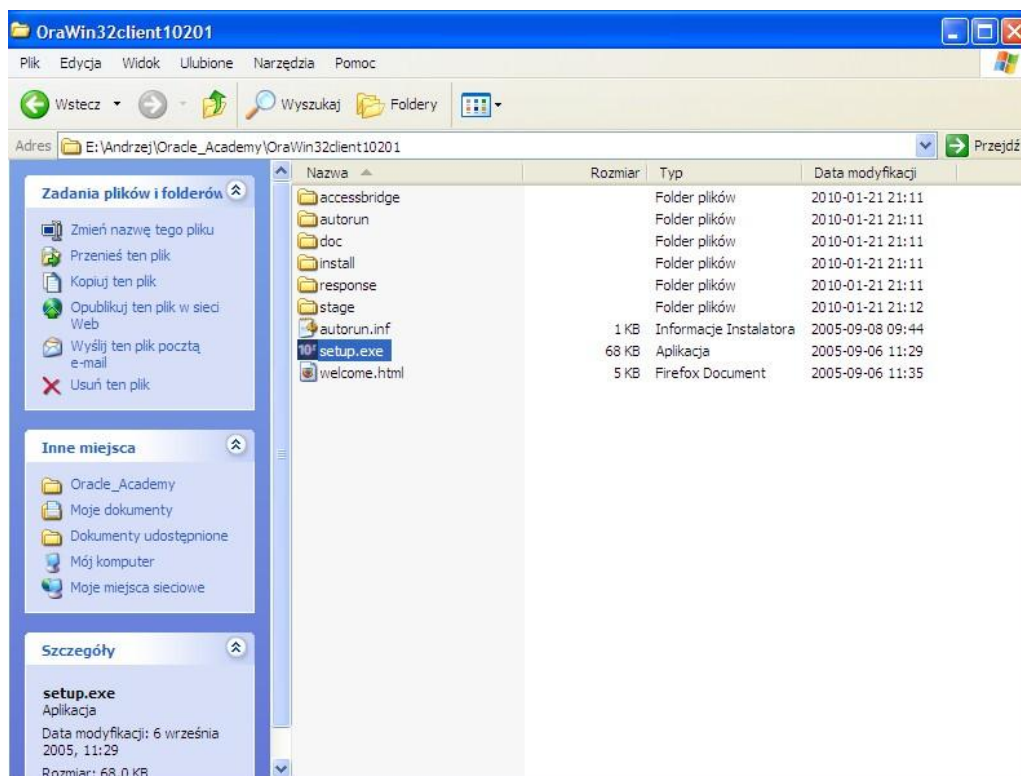
C. Instalacja i konfiguracja klienta serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional

Istnieje wiele wersji oprogramowania, które pełni rolę klienta bazy danych Oracle 10g. Przede wszystkim należy mieć świadomość, że w ramach instalacji systemu zarządzania bazą danych instalowane jest zarówno oprogramowanie serwera, jak i klienta. Dlatego nie jest konieczne odrębne instalowanie oprogramowania klienta na serwerze bazy danych. Jeśli jednak, pomimo że nie jest to zalecane, chcielibyśmy to uczynić z jakiegoś powodu, to powinniśmy przeprowadzić tę instalację w innym domu Oracle, czyli w innym katalogu niż ten, w którym zainstalowano system zarządzania bazą danych.

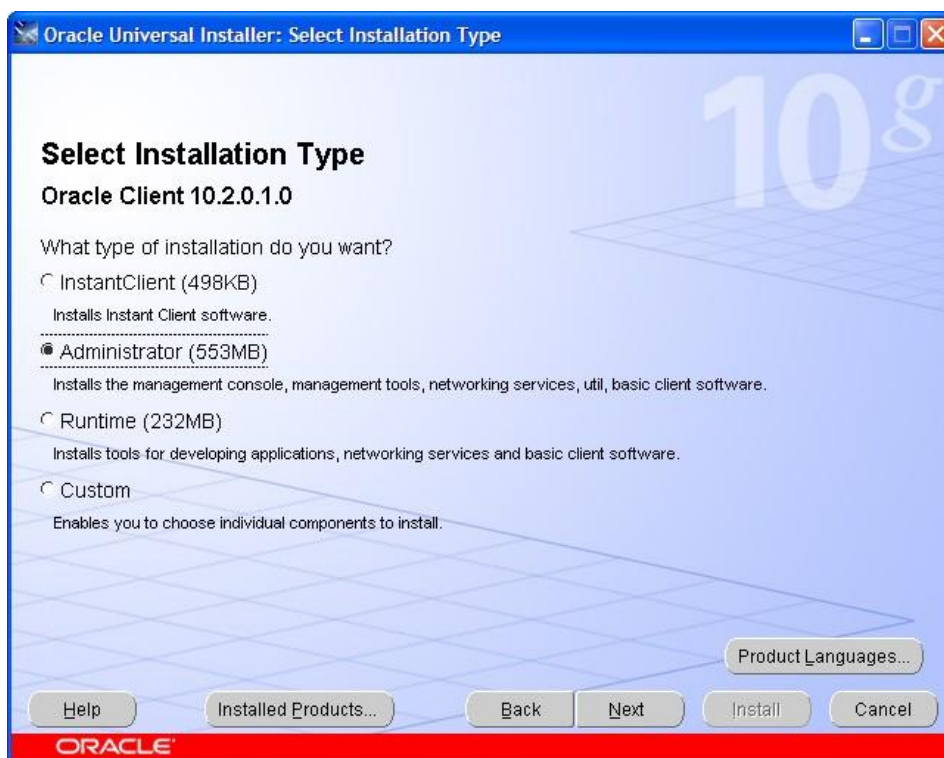
Jeśli chcemy zainstalować klienta bazy danych Oracle, powinniśmy sprawdzić wersję serwera bazy danych i zdobyć wersję oprogramowania klienta odpowiednią do tej bazy danych. Podobnie jak w przypadku systemu zarządzania bazą danych, należy zwrócić uwagę na rodzaj i wersję, w tym bitowość, systemu operacyjnego komputera, na którym to oprogramowanie ma być zainstalowane. Oprogramowanie instalacyjne klienta bazy danych, odpowiednie dla wersji serwera bazy danych, którego instalacja została opisana w dodatku A, jest dostępne pod tym samym adresem <http://www.oracle.com/technetwork/database/10201winsoft-095341.html>, pod którym znajduje się oprogramowanie instalacyjne systemu zarządzania bazą danych. Chociaż podane w tym akapicie uwagi są istotne, aby zapewnić pełną kompatybilność klienta i serwera

bazy danych, to należy przyznać, że bardzo często prosty, ale podstawowy program kliencki `sqlplus` potrafi poprawnie obsługiwać różne wersje bazy danych.

W pakiecie instalacyjnym mamy do dyspozycji oprogramowanie klienta serwera bazy danych (rys. C.1), które możemy zainstalować w różnych wersjach, możliwych do wyboru w początkowej fazie instalacji (rys. C.2). Aby zainstalować najpełniejszy zestaw oprogramowania dla klienta, należy oczywiście wybrać opcję *Administrator*. Natomiast opcja *InstantClient* umożliwia zainstalowanie najprostszego klienta serwera bazy danych Oracle, który jest np. potrzebny do połączenia się z poziomym skryptu PHP z bazą danych Oracle, co zostało opisane w dodatku D.



C.1 Rozpakowujemy pakiet instalacyjny klienta i uruchamiamy jego program `setup.exe`



C.2 Aby zainstalować najbogatszy zestaw oprogramowania dla klienta, należy wybrać opcję *Administrator*

Innym, mogącym nas interesować oprogramowaniem, które w dużej mierze ma charakter klienta bazy danych, jest Oracle Developer Suite 10g, dostępny pod adresem <http://www.oracle.com/technetwork/developer-tools/developer-suite/downloads/index.html> i dostosowany głównie do baz danych Oracle 10g, który zawiera narzędzia programistyczne do tworzenia oprogramowania głównie baz danych Oracle, a wśród nich pakiet Oracle Designer 10g, umożliwiający m. in. projektowanie i generowanie schematów baz danych Oracle. Przykładowo tworzenie projektów w Oracle Designer 10g wymaga wcześniejszego przygotowania odpowiedniego repozytorium w bazie danych Oracle. Natomiast w programie Oracle Designer 10g możemy pracować po zalogowaniu się na konto użytkownika repozytorium, czyli konto odpowiedniego użytkownika bazy danych, w której repozytorium dla Oracle Designer 10g zostało utworzone. W tym sensie program Oracle Designer 10g jest lokalnym bądź zdalnym klientem serwera bazy danych, w której zostało utworzone jego repozytorium.

Podstawową metodą autentykacji klienta serwera bazy danych Oracle jest tzw. autentykacja sieciowa (lub poprzez sieć) nawet jeśli klient i serwer bazy danych są uruchomione na tym samym komputerze, tzn. nawet jeśli są lokalnie dostępne w tym samym systemie. Dlatego niezależnie czy klient ma tylko zdalny dostęp do serwera bazy danych czy też lokalny, należy go tak skonfigurować, aby pomyślnie przechodził weryfikację połączeń z żądanym serwerem bazy danych na podstawie autentykacji sieciowej. Możemy to zrobić korzystając z narzędzia Oracle Net Configuration Assistant, którego wykorzystanie w pewnym zakresie na serwerze bazy danych zostało opisane w dodatku A. Instalacje oprogramowania klienta zwykle kończą się uruchomieniem właśnie tego programu. Jednak poniżej zostanie pokazane, gdzie należy go szukać, gdy proces instalacji został już zakończony. Zostaną uwzględnione dwa przypadki: klient z pakietu instalacyjnego Oracle Database 10g oraz pakiet Oracle Developer Suite 10g.

Załóżmy, że oprogramowanie klienta z pakietu instalacyjnego Oracle Database 10g zostało zainstalowane w domu Oracle o nazwie OraDb10g_clienthome1 w ścieżce `d:\oracle\product\10.2.0\client_1`. Wówczas Oracle Net Configuration Assistant będzie można uruchomić z poziomu menu *Start* → *Oracle* – *OraDb10g_clienthome1* → *Configuration and Migration Tools* → *Net Configuration Assistant*, czyli bardzo podobnie jak zostało to pokazane w przypadku serwera na rys. A.28. W celu uruchomienia Oracle Net Configuration Assistant można też wykonać plik wsadowy `d:\oracle\product\10.2.0\client_1\bin\netca.bat`.

Załóżmy natomiast, że oprogramowanie pakietu instalacyjnego Oracle Developer Suite 10g zostało zainstalowane w domu Oracle o nazwie DevSuiteHome1 w ścieżce `d:\oracle\product\DevSuiteHome_1`. Wówczas Oracle Net Configuration Assistant z pakietu Oracle Developer Suite 10g będzie można uruchomić z poziomu menu *Start* → *Oracle* – *DevSuiteHome1* → *Configuration and Migration Tools* → *Net Configuration Assistant*. W celu uruchomienia Oracle Net Configuration Assistant z pakietu Oracle Developer Suite 10g można też wykonać plik wsadowy `d:\oracle\product\DevSuiteHome1\bin\netca.bat`.

Należy pamiętać, że ustalając parametry tzw. *lokalnych nazw usług sieciowych* (*Local Net Service Name*), które najczęściej są używane do konfiguracji połączeń sieciowych z serwerem bazy danych za pomocą Oracle Net Configuration Assistant, będą domyślnie zapisywane w tym domu Oracle, z którego został uruchomiony program Oracle Net Configuration Assistant. Parametry potrzebne dla ustawienia połączenia klienta z określonym serwerem bazy danych za pomocą autentykacji sieciowej są zapisywane do pliku `tnsnames.ora`, którego lokalizacje przy przyjętych powyżej założeniach będą następujące:

- dla klienta z pakietu Oracle Database 10g:

```
d:\oracle\product\10.2.0\client_1\NETWORK\ADMIN\tnsnames.ora
```

- dla klienta z pakietu Oracle Developer Suite 10g:

```
d:\oracle\product\DevSuiteHome_1\NETWORK\ADMIN\tnsnames.ora
```

Jeśli jednak w systemie operacyjnym zdefiniowana będzie systemowa zmienna środowiskowa ORACLE_HOME, to domyślnym plikiem tnsnames.ora będzie ten, który znajduje się w ścieżce: ORACLE_HOME\NETWORK\ADMIN\tnsnames.ora. Przykładowo, jeśli

ORACLE_HOME = d:\oracle\product\DevSuiteHome_1, to zmiany konfiguracji *lokalnej usługi nazwy* dla połączenia sieciowego wykonane za pomocą Oracle Net Configuration Assistant z klienta pakietu Oracle Database 10g także będą zapisywane do pliku:

```
d:\oracle\product\DevSuiteHome_1\NETWORK\ADMIN\tnsnames.ora.
```

Natomiast definiując zmienną środowiskową TNS_ADMIN dla danego użytkownika systemu operacyjnego możemy określić lokalizację pliku tnsnames.ora, np. tak jak zostało to pokazane w dodatku D, która będzie nadrzędna dla wszystkich klientów serwera bazy danych Oracle, zainstalowanych na koncie tego użytkownika systemu operacyjnego.

Na rys. C.3 – C.11 zostały pokazane kolejne etapy konfigurowania *lokalnej nazwy sieciowej usługi bazy danych* przy użyciu programu Oracle Net Configuration Assistant. Aby można było przeprowadzić taką konfigurację, muszą być znane następujące parametry:

- Nazwa usługi bazy danych, która najczęściej jest taka sama jak globalna bazy danych. W naszym przykładzie jest to nazwa domenowa: `umain.pkif.us.edu.pl`
- Nazwa domenowa komputera, na którym zainstalowano serwer bazy danych. W naszym przykładzie jest to `oracle1.pkif.us.edu.pl`
- Port, na którym serwer bazy danych prowadzi nasłuch sieciowy. Domyślnie jest to port 1521, który także został wykorzystany w naszym przykładzie.

Wymienione powyżej parametry wchodzi w skład deskryptora połączenia, który charakteryzuje ustaloną (w naszym przykładzie na rys. 11) nazwę sieciową `umain` usługi bazy danych. Po przeprowadzeniu konfiguracji pokazanej na rys. C.3 – C.11 w odpowiednim pliku `tnsnames.ora` znajduje się nazwa sieciowa usługi bazy danych wraz z jej deskrytorem połączenia. Zawartość przykładowego pliku `tnsnames.ora` w naszym przykładzie będzie następująca:

```

# tnsnames.ora Network Configuration File:
D:\oracle\product\10.2.0\client_1\network\admin\tnsnames.ora
# Generated by Oracle configuration tools.

UMAIN =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = oracle1.pkif.us.edu.pl) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = umain.pkif.us.edu.pl)
    )
  )

```

Należy tutaj podkreślić, że plik `tnsnames.ora`, podobnie jak `listener.ora` i `sqlnet.ora`, które zostały opisane w dodatku A, jest plikiem tekstowym i w związku z tym możemy dokonywać w nim zmian w dowolnym edytorze tekstowym bez pomocy Oracle Net Configuration Assistant. Jednak szczególnie dla początkujących użytkowników systemów baz danych Oracle wygodne i zalecane jest zastosowanie tego narzędzia do konfigurowania parametrów połączeń z serwerami baz danych Oracle.



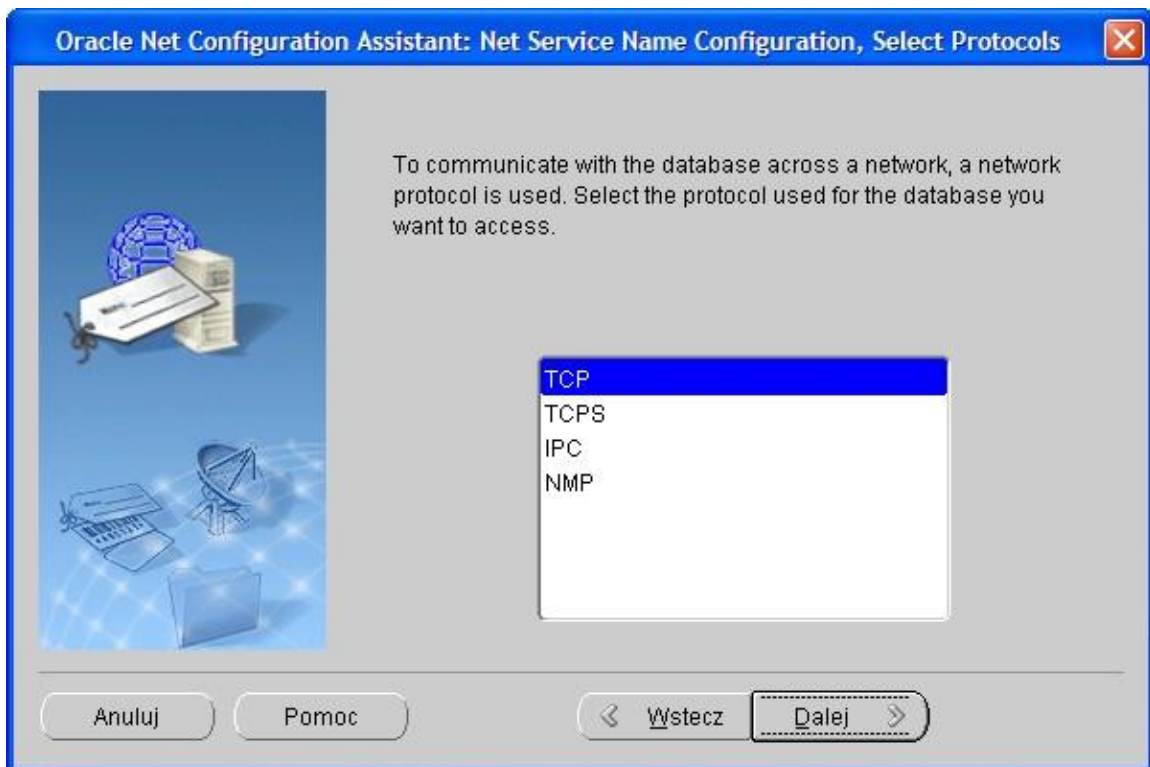
C.3 Wybieramy konfigurację lokalnej nazw sieciowej usługi bazy danych, z którą chcemy się połączyć



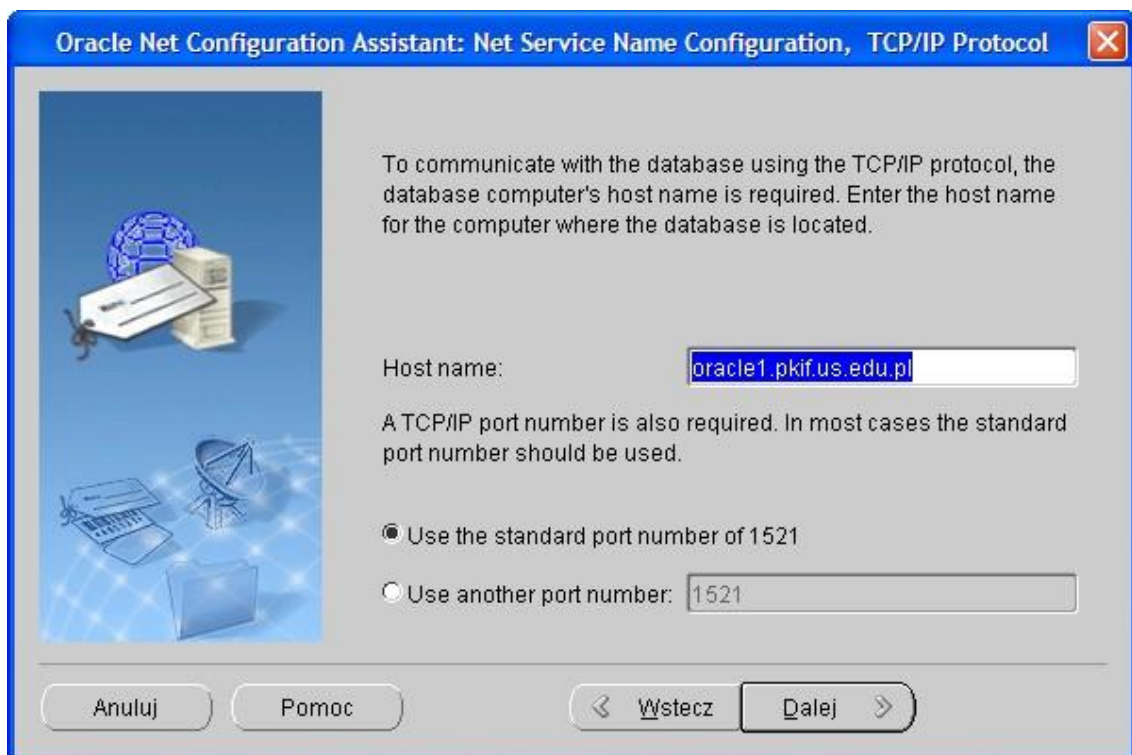
C.4 Rozpoczynamy dodawanie nazwy sieciowej usługi bazy danych



C.5 Podajemy nazwę usługi bazy danych, która zwykle jest też globalną nazwą bazy danych



C.6 Wybieramy protokół sieciowy, zwykle TCP, który wykorzystamy do połączenia z bazą danych



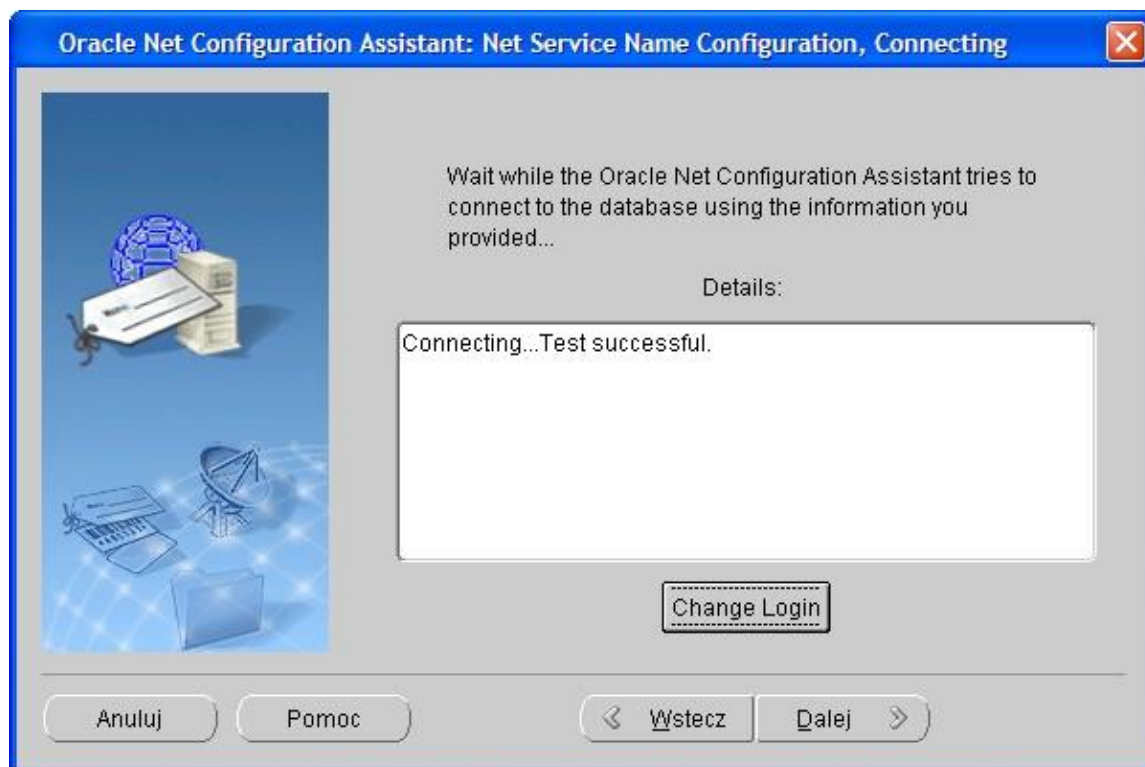
C.7 Podajemy domenową nazwę komputera, na którym znajduje się serwer bazy danych



C.8 Możemy przeprowadzić test połączenia



C.9 Typowym problemem podczas testu dobrze skonfigurowanego połączenia jest niewłaściwe hasło użytkownika SYSTEM bazy danych, które należy zmienić i ponownie test połączenia



C.10 Po zmianie hasła użytkownika SYSTEM bazy danych test połączenia przechodzi pomyślnie



C.11 Ustalamy nazwę sieciowej usługi bazy danych, którą klient będzie używać do połączeń z serwerem bazy danych

Po zdefiniowaniu nazwy sieciowej usługi bazy danych możemy zastosować ją do połączeń z serwerem bazy danych. Na tym w praktyce polega łączenie się klienta z serwerem na podstawie autentykacji sieciowej. Przykładowo stosując zdefiniowaną na komputerze klienta nazwę sieciową umain usługi bazy danych `umain.pkif.us.edu.pl`, możemy połączyć się z komputera klienta z użytkownikiem SYSTEM tej bazy danych np. za pomocą podstawowego terminala bazy danych Oracle SQL*Plus, wpisując w konsoli systemu operacyjnego komputera klienta komendę:

```
sqlplus system@umain
```

Powyższa komenda zadziała poprawnie przy dobrze zdefiniowanej zmiennej środowiskowej ORACLE_HOME w systemie operacyjnym na komputerze klienta. Jeśli nie taka zmienna nie została zdefiniowana, to programu sqlplus należy szukać wśród innych programów wykonywalny w katalogu, w którym został zainstalowany dany pakiet aplikacji Oracle. W przypadku rozważonych powyżej pakietów narzędzi Oracle będą to:

- dla klienta z pakietu Oracle Database 10g:

```
d:\oracle\product\10.2.0\client_1\bin\sqlplus.exe
```

- dla klienta z pakietu Oracle Developer Suite 10g:

```
d:\oracle\product\DevSuiteHome_1\bin\sqlplus.exe
```

W środowisku systemu operacyjnego Windows mamy także do dyspozycji graficzną wersję tego terminala, dostępną z poziomu menu *Start*, przykładowo:

- dla klienta z pakietu Oracle Database 10g:

Start → *Oracle – OraDb10g_clienthome1* → *Application Development* → *SQL Plus*

- dla klienta z pakietu Oracle Developer Suite 10g:

Start → *Oracle – DevSuiteHome1* → *Application Development* → *SQL Plus*.

W ten sposób uruchamiamy program `sqlplusw`, który w przypadku naszych przykładowych instalacji:

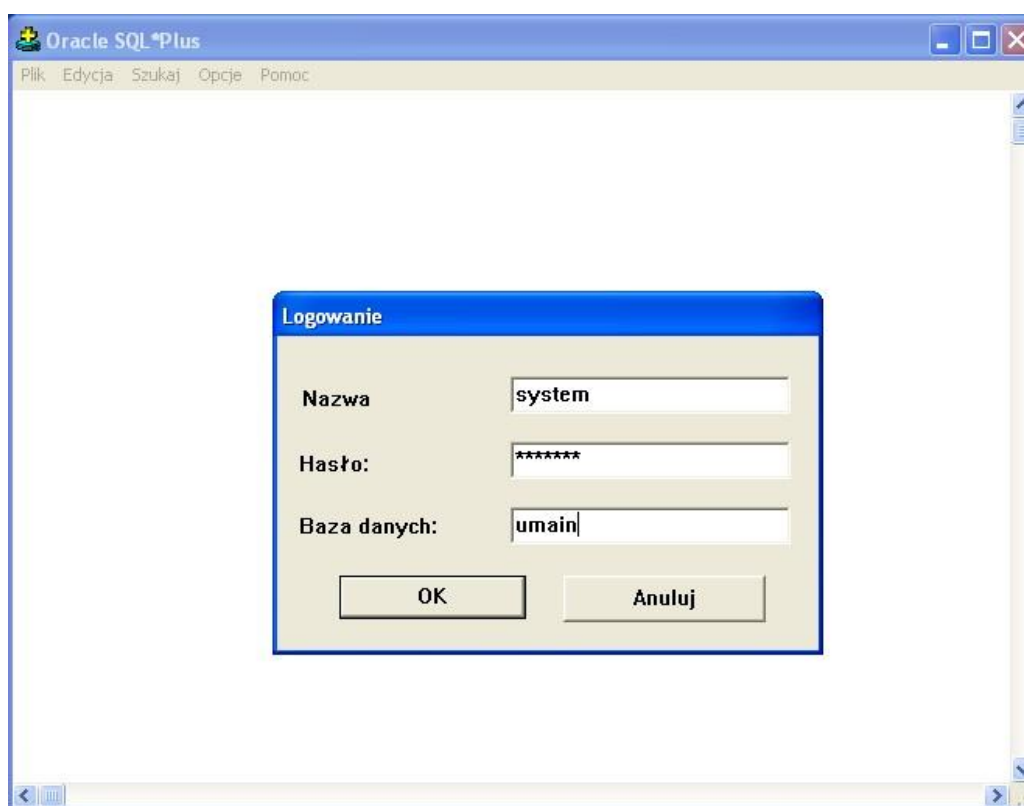
- dla klienta z pakietu Oracle Database 10g znajduje się w ścieżce:

d:\oracle\product\10.2.0\client_1\bin\sqlplusw.exe

- dla klienta z pakietu Oracle Developer Suite 10g jest dostępny w ścieżce:

d:\oracle\product\DevSuiteHome_1\bin\sqlplusw.exe

Wówczas na konto użytkownika SYSTEM bazy danych `umain.pkif.us.edu.pl` o lokalnej nazwie sieciowej `umain` jej usługi logujemy się w sposób podany na rys. C.12, oczywiście podając odpowiednie hasło dla tego użytkownika administracyjnego bazy danych.



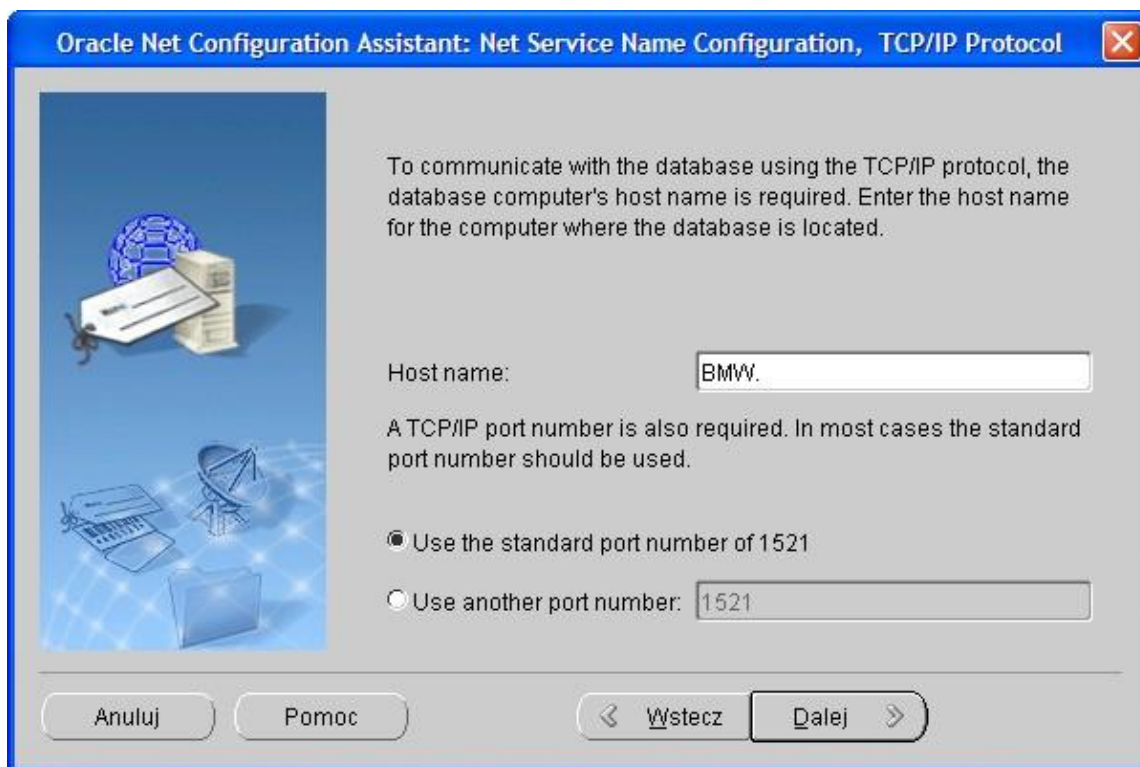
C.12 Logowanie się na konto użytkownika SYSTEM bazy danych o lokalnej nazwie sieciowej `umain` zdefiniowanej w odpowiednim pliku `tnsnames.ora`

Kończąc temat wprowadzenia do autentykacji sieciowej klienta serwera bazy danych Oracle, należy zaznaczyć, że również na komputerze pełniącym rolę serwera bazy danych warto zdefiniować nazwę sieciowej usługi bazy danych. Umożliwia to łączenie się na takim komputerze ze znajdującą się na nim bazą danych na podstawie autentykacji sieciowej, która jest stosunkowo łatwa do skonfigurowania i najczęściej wymagana przez różne narzędzia

Oracle. Aby przeprowadzić określenie i poprawnie skonfigurować nazwę sieciowej usługi bazy danych lokalnie na serwerze bazy danych Oracle, także możemy skorzystać z programu Oracle Net Configuration Assistant. W tym celu, biorąc jako przykład instalację systemu zarządzania bazą danych Oracle opisaną w dodatku A, możemy uruchomić Oracle Net Configuration Assistant z poziomu menu *Start* → *Oracle – OraDb10g_dbhome1* → *Configuration and Migration Tools* → *Net Configuration Assistant* bądź też wykonując plik wsadowy `d:\oracle\product\10.2.0\db_1\bin\netca.bat`. Wówczas będziemy mogli powtórzyć kroki konfiguracji pokazane na rys. C.3 – C.11. Jednak chcąc określić deskryptor połączenia do bazy `edudb` z dodatku A, kroki konfiguracji pokazane na rys. C.5 i C.7 należy zastąpić odpowiednio tymi, które przedstawiono na rys. C.13 i C.14.



C.13 Podajemy nazwę usługi bazy danych, która jest też globalną nazwą bazy danych `edudb`



C.14 Podajemy domenową nazwę komputera, na którym znajduje się serwer bazy danych. W przypadku bazy danych edudb, opisanej w dodatku A, domena komputera, na którym zainstalowano serwer bazy danych jest pusta.

W rezultacie deskryptor połączenia zapisany w odpowiednim pliku tnsnames.ora (w przypadku instalacji serwera bazy danych edudb opisanej w dodatku A będzie to plik D:\oracle\product\10.2.0\db_1\network\admin\tnsnames.ora) dla utworzonej lokalnej nazwy sieciowej edudb usługi bazy danych o tej samej nazwie jest następujący:

```
EDUDB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = BMW.) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = edudb)
    )
  )
```

Wówczas będziemy mogli łączyć się z bazą danych edudb lokalnie z poziomu komputera, na którym ta baza danych się znajduje, na podstawie autentykacji sieciowej, używając nazwy sieciowej edudb usługi bazy danych o tej samej nazwie, np. wydając w konsoli systemu operacyjnego komputera klienta komendę:

```
sqlplus system@edudb
```

Oczywiście powyższe polecenie zostanie zrealizowane bez podania pełnej ścieżki dostępu do programu `sqlplus.exe`, jeśli w systemie operacyjnym będzie poprawnie zdefiniowana zmienna środowiskowa `ORACLE_HOME`. W przypadku instalacji opisanej w dodatku A zmienna ta powinna być określona w następujący sposób:

```
ORACLE_HOME = d:\oracle\product\10.2.0\db_1
```

D. Konfiguracja klienta serwera bazy danych Oracle 10g w systemie operacyjnym Windows XP Professional dla połączeń skryptów PHP z bazą danych

Jednym z powszechnie stosowanych języków skryptowych do tworzenia internetowych aplikacji współpracujących z bazami danych jest PHP (*PHP: Hypertext Preprocessor*), posiadający duży zestaw rozszerzeń, które umożliwiają wygodny dostęp do baz danych wielu różnych producentów. Popularnym rozszerzeniem PHP obsługującym bazy danych Oracle jest `PHP_OCI8`. Jednak jeśli nasze skrypty PHP mają łączyć się z serwerem bazy danych Oracle 10g nie wystarczy jedynie uaktywnić to rozszerzenie (jak było to np. w przypadku serwera bazy danych Oracle 8i) poprzez dodania w sposób pokazany poniżej w punkcie 1 odpowiedniego wpisu do pliku `php.ini` serwera WWW, który daje możliwość uruchamiania skryptów PHP. Chcąc w wygodny sposób definiować w skryptach PHP połączenia z bazą danych, należy jeszcze zainstalować i skonfigurować prostego klienta (*Oracle Instant Client*) serwera bazy danych Oracle 10g na komputerze, na którym został postawiony serwer WWW z PHP. Poniżej znajduje się szczegółowy opis, jak przeprowadzić całą tę procedurę w przypadku, gdy serwer WWW jest uruchomiony w systemie Windows XP Professional:

1. Udostępnić w pliku PHP.INI rozszerzenie PHP_OCI8 poprzez aktywną komendę:

```
extension=php_oci8.dll
```

i zrestartować serwer Apache.

2. Zainstalować *Oracle Instant Client* (dokładnie odpowiadający wersji serwera Oracle, z którym nasze skrypty PHP mają się łączyć) w wybranym katalogu, np.

```
c:\instantclient10_2
```

- najlepiej z pakietu instalacyjnego (rys. C. 1), wybierając opcję instalacji najprostszego klienta, którego nazwano *Instant Client* (rys. C. 2)

- ewentualnie ściągnąć potrzebny zestaw bibliotek ze strony:

<http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html>

3. Usunąć lub zmienić nazwę biblioteki *oraoci10.dll*, znajdującej się w katalogu, w którym zainstalowano *Instant Client*.

4. Dodać do zmiennej systemowej *Path* ścieżkę dostępu do katalogu, w którym zainstalowany został *Instant Client*.

Np. w systemie Windows XP PL można to zrobić w:

Start → Panel sterowania → Wydajność i konserwacja → System → Zaawansowane → Zmienne środowiskowe → Zmienne systemowe

czyli w naszym przykładzie zmienna systemowa *Path* = *c:\instantclient10_2;...*

5. Przegrać plik *tnsnames.ora*, znajdujący się w podkatalogu instalacyjnym serwera

ORACLE_HOME\NETWORK\ADMIN (czyli w przypadku naszej przykładowej bazy danych z dodatku A jest to katalog *d:\oracle\product\10.2.0\db_1\NETWORK\ADMIN*),

do katalogu, w którym został zainstalowany *Instant Client* (w naszym przykładzie do

c:\instantclient10_2), a jeśli go tam nie ma, to utworzyć ten plik np. za pomocą

Oracle Network Configuration Assistant, podając odpowiednie parametry serwera bazy danych, podobnie jak to zrobiono w dodatku C.

6. Ustawić zmienną środowiskową użytkownika

```
TNS_ADMIN=c:\instantclient10_2
```

Np. w systemie Windows XP PL można to zrobić w:

Start → Panel sterowania → Wydajność i konserwacja → System → Zaawansowane →
Zmienne środowiskowe → Zmienne użytkownika dla nazwa_użytkownika

Opcjonalnie można też ustawić zmienną środowiskową użytkownika

LOCAL = sieciowa nazwa bazy danych zdefiniowana w pliku
tnsnames.ora

np. LOCAL = edudb

Spowoduje to automatyczne logowanie się do tej instancji bazy danych (w naszym przykładzie edudb) za pomocą funkcji `oci8logon` rozszerzenia PHP_OCI8 bez podawania jej trzeciego argumentu, czyli sieciowej nazwy bazy danych.

E. Wybrane podstawowe operacje administracyjne – czyli co warto wiedzieć, gdy rozpoczynamy pracę na serwerze bazy danych Oracle 10g

1. Warto ustawić systemowe zmienne środowiskowe `ORACLE_HOME` i `ORACLE_SID`, aby mieć wygodny dostęp do administracyjnych programów narzędziowych zgromadzonych w katalogu `ORACLE_HOME\bin` oraz wskazać w `ORACLE_SID` domyślną bazę danych, do której będą odwoływać się różne operacje, jak np. start konsoli *Enterprise Manager* naszej bazy danych. W systemie operacyjnym Windows XP PL możemy to zrobić poprzez:

Start → Panel sterowania → Wydajność i konserwacja → System → Zaawansowane →
Zmienne środowiskowe → Zmienne systemowe

Dla naszej bazy z dodatku A powinniśmy ustawić:

`ORACLE_HOME = D:\oracle\product\10.2.0\db_1`

`ORACLE_SID = edudb`

2. Konwersja ustawień narodowych (w tym zestawu znaków) po stronie klienta:

Wystarczy odpowiednio ustawić zmienną środowiskową `NLS_LANG`, np. w systemie operacyjnym Windows poprzez wydanie w konsoli systemu Windows poniższej komendy

```
set NLS_LANG=POLISH_POLAND.EE8PC852
```

Po tym po uruchomieniu w tej samej konsoli podstawowego programu terminalowego bazy danych Oracle, tj. programu `sqlplus`, będą poprawnie wyświetlane polskie znaki diakrytyczne.

3. Zmianianie/tworzenie pliku haseł użytkowników administracyjnych (z rolami DBA lub DBAOPER) – podczas tej operacji baza powinna być zamknięta

- w systemie Windows

```
orapwd file=PWD<SID> password=nowe_hasło entries=5 force=y
```

- w systemie Linux

```
orapwd file=orapw<SID> password=nowe_hasło entries=5 force=y
```

Te komendy utworzą nowy plik haseł dla administracyjnego użytkownika SYS naszej bazy danych: w systemie Windows plik `PWD<SID>` w domyślnej lokalizacji `ORACLE_HOME\database` oraz w systemie Linux plik `orapw<SID>` w domyślnej lokalizacji `ORACLE_HOME/dbs`. W nazwie pliku haseł zamiast `<SID>` powinniśmy wpisać identyfikator instancji odpowiedniej bazy danych – w naszym przykładzie z dodatku A jest to `<SID> = edudb`.

4. Rodzaje logowania

`sqlplus /nolog` – uruchomienie w konsoli systemu operacyjnego programu `sqlplus` bez logowania

Następnie mamy kilka możliwości logowania za pomocą komendy `connect` w środowisku programu `sqlplus`:

`connect /as sysdba` – autentykacja systemowa (systemu operacyjnego) dla użytkownika administracyjnego bazy danych – możliwa jedynie z poziomu systemu operacyjnego, na którym postawiono serwer bazy danych – nie korzysta z pliku haseł; Warto wiedzieć, że ten sam rezultat, jaki dają poprzednie dwie komendy, można osiągnąć jako użytkownik systemu operacyjnego będącego administratorem bazy danych Oracle (w

systemach Linux lub Unix jest to zwykle użytkownik oracle) za pomocą jednego polecenia konsoli systemu operacyjnego: `sqlplus / as sysdba`

`connect sys as sysdba` – logowanie głównego użytkownika administracyjnego bazy danych dzięki ustalonej zmiennej środowiskowej `ORACLE_SID` – możliwe jedynie z poziomu systemu operacyjnego, na którym postawiono serwer bazy danych – korzysta z pliku haseł;

`connect user/password` – logowanie zwykłego, nieadministracyjnego użytkownika bazy danych dzięki ustalonej zmiennej środowiskowej `ORACLE_SID` – możliwa jedynie z poziomu systemu operacyjnego, na którym postawiono serwer bazy danych – nie korzysta z pliku haseł;

`connect sys@edudub as sysdba` – logowanie głównego użytkownika administracyjnego bazy danych dzięki autentykacji sieciowej (poprzez deskryptor połączenia o nazwie `edudb` zdefiniowany w dodatku C jako sieciowa nazwa naszej bazy danych z dodatku A, który powinien być zapisany na komputerze klienta w pliku `ORACLE_HOME\NETWORK\ADMIN\tnsnames.ora` w systemie Windows lub w `ORACLE_HOME/Network/admin/tnsnames.ora` w systemie Linux) – korzysta z pliku haseł;

`connect user/password@edudb` – logowanie zwykłego, nieadministracyjnego użytkownika bazy danych dzięki autentykacji sieciowej (poprzez deskryptor połączenia o nazwie `edudb` zdefiniowany w dodatku C jako sieciowa nazwa naszej bazy danych z dodatku A, który powinien być zapisany na komputerze klienta w pliku `ORACLE_HOME\NETWORK\ADMIN\tnsnames.ora` w systemie Windows lub w `ORACLE_HOME/Network/admin/tnsnames.ora` w systemie Linux) – nie korzysta z pliku haseł.

5. Podstawy obsługi procesu nasłuchu sieciowego serwera bazy danych w konsoli systemu operacyjnego serwera bazy danych:

`lsnrctl status` – sprawdzenie statusu procesu nasłuchu sieciowego (listenera)

```
lsnrctl start – start listenera  
lsnrctl stop – zatrzymanie listenera
```

Powyższe komendy z opcjami `start` i `stop` powodują odpowiednio uruchomienie i zatrzymanie usługi procesu nasłuchu serwera, co w systemie operacyjnym Windows jest też możliwe za pomocą menadżera usług lub przy użyciu komendy `sc`, jak to zostało przedstawione w dodatku B.

6. Etapy startowania instancji bazy danych Oracle (np. w narzędziu `sqlplus`) po zalogowaniu na konto głównego użytkownika administracyjnego (SYS)

- Uruchomienie bazy danych bez montowania, tzn. w trybie `nomount` – skutkuje uruchomieniem procesów instancji bazy danych, działających w tle i odczyt pliku parametrów (najpierw następuje próba odczytu dynamicznego pliku parametrów `spfile<SID>.ora` w domyślnej lokalizacji `ORACLE_HOME\dbs` w systemie Windows lub `ORACLE_HOME/dbs` w systemie Linux, a jeśli nie zostanie on znaleziony, to jest odczytywany statyczny plik parametrów `init.ora`).

Komenda w `sqlplus`:

```
startup nomount
```

Następnie, aby przejść do kolejnych etapów otwierania bazy, kolejno:

```
alter database mount;
```

```
alter database open;
```

- Zamontowanie bazy danych, tzn. otwarcie w trybie `mount` – poza tym co jest wykonywane w trybie `nomount` powoduje otwarcie i odczyt plików sterujących bazą danych (`controlfiles`)

Komenda w `sqlplus`:

```
startup mount
```

Następnie, aby ostatecznie otworzyć bazę:

```
alter database open;
```

- Pełne otwarcie bazy danych (tryb `open`)

Komendy w `sqlplus`:

startup

lub

startup open

W przypadku zastosowania jednej z powyższych komend w pełni otwierających bazę danych Oracle wykonywane są kolejne operacje związane z trybami nomount i mount, a następnie otwierane są plik danych bazy danych i inne pliki potrzebne do obsługi danych w bazie, np. pliki dziennika powtórzeń.

7. Rodzaje zamykania instancji bazy Oracle (np. w narzędziu sqlplus) po zalogowaniu na konto głównego użytkownika administracyjnego (SYS)

shutdown normal – zamknięcie dopiero po rozłączeniu wszystkich użytkowników

shutdown transactional – zamknięcie po zakończeniu wszystkich rozpoczętych wcześniej transakcji

shutdown immediate – zamknięcie jest poprzedzone wycofywaniem wszystkich rozpoczętych, a jeszcze nieukończonych transakcji – jest to najwłaściwsza forma, aby w szybki i spójny sposób zamknąć bazę danych.

shutdown abort – brutalne (wysoce niezalecane) zamknięcie bazy, które jest uzasadnione tylko w przypadku poważnych awarii (jak utrata dysku z niektórymi plikami bazy danych)

8. Podstawy obsługi zdalnej administracyjnej konsoli bazy danych (pozwalającej korzystać z narzędzia *Oracle Enterprise Manager* poprzez przeglądarkę internetową):

emctl status dbconsole – sprawdzenie statusu konsoli *Oracle Enterprise Manager* bazy danych

emctl start dbconsole – start konsoli *Oracle Enterprise Manager* bazy danych

emctl stop dbconsole – zatrzymanie konsoli *Oracle Enterprise Manager* bazy danych

Powyższe komendy z opcjami start i stop powodują odpowiednio uruchomienie i zatrzymanie usługi konsoli *Oracle Enterprise Manager* bazy danych, co w systemie

operacyjnym Windows jest też możliwe za pomocą menadżera usług lub przy użyciu komendy `sc`, jak to zostało przedstawione w dodatku B.

9. Skrót informacji dotyczących przywilejów i ról (zbiorów przywilejów), przyznawanych użytkownikom bazy danych.

Nowy użytkownik utworzony z poziomu użytkownika administracyjnego (np. `SYS` lub `SYSTEM`), np. w narzędziu `sqlplus` za pomocą komendy:

```
create user nazwa_uzytkownika identified by haslo_uzytkownika;
```

nie będzie mógł niczego zrobić, nawet nie będzie mógł zalogować się do bazy.

W bazach danych Oracle istnieją dwie grupy przywilejów **systemowe** (np. `create table`, `create view`) i **obiektowe** (np. `select`, `insert`, `update`, `delete on nazwa_tabeli`). Ponadto można tworzyć role, zawierające zestawy przywilejów.

Przykładowo nowoutworzonemu użytkownikowi można nadać predefiniowane, standardowe role:

`CONNECT`, która w Oracle 10g zawiera tylko jeden przywilej systemowy `CREATE SESSION` oraz

`RESOURCE`, która zawiera przywileje systemowe `CREATE CLUSTER`, `CREATE INDEXTYPE`, `CREATE OPERATOR`, `CREATE PROCEDURE`, `CREATE SEQUENCE`, `CREATE TABLE`, `CREATE TRIGGER`, `CREATE TYPE`.

Robimy to poleceniem:

```
grant connect, resource to nazwa_uzytkownika;
```

Zauważmy, że w Oracle 10g rola `RESOURCE` nie posiada przywileju `CREATE VIEW`, który należałoby nadać dodatkowo, aby posiadał on prawo tworzenia własnych perspektyw, np. rozszerzając poprzednią komendą do następującej postaci:

```
grant connect, resource, create view to nazwa_uzytkownika;
```

Natomiast przykładem nadania przywilejów obiektowych, może być nadanie użytkownikowi u1 praw `select`, `insert`, `update`, `delete` do tabeli `emp` użytkownika `scott` za pomocą komendy:

```
grant select, insert, update, delete on scott.emp to u1;
```

Przywileje i role odbieramy komendą `revoke`, np.:

```
revoke delete on scott.emp from u1;
```

Przywileje i role systemowe możemy nadawać z lub bez opcji `WITH ADMIN OPTION`, natomiast przywileje obiektowe mogą być przyznawane z lub bez opcji `WITH GRANT OPTION`. Jednak należy mieć świadomość, że w istotnym aspekcie bezpieczeństwa dostępu do danych składowanych w bazie działanie tych opcji jest inne.

Przypadek przywileju obiektowego:

```
grant select on scott.dept to u1; – wówczas u1 nie może przekazać tego  
przywileju innemu użytkownikowi
```

```
grant select on scott.dept to user1 with grant option;
```

– w tym przypadku `user1` może przekazać ten przywilej innemu użytkownikowi, np. użytkownikowi `user2`, wykonując komendę:

```
grant select on scott.dept to user2;
```

Wówczas komenda:

```
revoke select on scott.dept from user1;
```

odbiera przywilej `select` on `scott.dept` zarówno użytkownikowi `user1`, jak i `user2`.

Przypadek przywileju systemowego:

W przypadku przywileju systemowego (np. `create view`) opcja `with admin option` komendy `grant` działa tak samo jak dla przywileju obiektowego opcja `with grant option` komendy `grant`. Natomiast komenda `revoke` daje niestety inny efekt, gdy obieramy przywilej systemowy użytkownika, który otrzymał go z opcją `with admin option`, a następnie nadał go innym użytkownikom.

Przykładowo komenda:

```
grant create view to user1 with admin option;
```

powoduje, że `user1` może nadać przywilej systemowy `create view` użytkownikowi `user2` za pomocą komendy:

```
grant create view to user2;
```

Wówczas komenda:

```
revoke create view from user1;
```

odbiera przywilej `create view` użytkownikowi `user1`, ale nie odbiera go użytkownikowi `user2`.

Aby odebrać przywilej `create view` użytkownikowi `user1`, należy wydać komendę:

```
revoke create view from user2;
```

Należy pamiętać, że przywileje i role systemowe nadane przez użytkownika, które otrzymał je z opcją `WITH ADMIN OPTION` nie mogą być odebrane kaskadowo.

Natomiast przeciwnie jest w przypadku przywilejów obiektowych nadanych przez użytkownika, który otrzymał je z opcją `WITH GRANT OPTION`. Są one odbierane kaskadowo innym użytkownikom, gdy odbieramy je użytkownikowi, który je im nadał.

10. Krótka informacja na temat perspektyw i tabel słownika danych bazy Oracle

Słownik bazy danych Oracle składa się zarówno z obiektów statycznych, jak i dynamicznych. W obydwu tych grupach znajdują się tabele i perspektywy. Wśród obiektów statycznych mamy też inne rodzaje obiektów, np. synonimy. Jednak głównie korzystamy z perspektyw słownika danych, gdy chcemy uzyskać informacje o stanie bazy danych i jej obiektach. Statyczne perspektywy słownika danych bazy Oracle dzielimy na trzy podstawowe kategorie, a przynależność do każdej z nich jest odzwierciedlona w nazwach perspektyw, które posiadają następujące przedrostki:

`user_` – informacje o obiektach bieżącego użytkownika

`all_` – informacje o obiektach bazy dostępne także dla nieadministracyjnych użytkowników

`dba_` – informacje o wszystkich obiektach dostępne wyłącznie dla użytkowników administracyjnych

Natomiast dynamiczne perspektywy słownika danych, umożliwiające monitorowanie w czasie rzeczywistym statystyk działania bazy danych, mają nazwy zaczynające się od przedrostka `v$`, a informacje w nich zawarte są dostępne wyłącznie dla użytkowników o uprawnieniach administracyjnych, np. posiadających rolę DBA.

Ogólna informacja o wszystkich statycznych obiektach słownika danych znajduje się w perspektywie `DICTIONARY`:

```
-----  
desc DICTIONARY  
select * from DICTIONARY;  
-----
```

Ogólna informacja o wszystkich dynamicznych obiektach słownika danych jest dostępna w perspektywie `V$FIXED_TABLE`:

```
-----  
desc V$FIXED_TABLE  
select * from V$FIXED_TABLE;  
-----
```

Można też korzystać z perspektywy `DICT_COLUMNS`, zawierającej informacje o kolumnach obiektów słownika danych:

```
select * from DICT_COLUMNS
where table_name='DBA_USERS';
```

Przydatne będą również perspektywy `ALL_VIEWS` oraz `DBA_VIEWS`.

Aby efektywnie korzystać z perspektyw słownika danych, najpierw należy poznać atrybuty danej perspektywy, np. za pomocą komendy `desc`, a później zadawać przemyślane zapytania, które dostarczą nam informację w przejrzysty sposób. W tym celu należy wybierać tylko konieczną ilość atrybutów danej perspektywy i konstruować właściwe warunki selekcji w klauzeli `where` zapytania SQL do tej perspektywy. Przykładowo, aby dowiedzieć się jakie tabele posiada użytkownik `SCOTT`, gdy jesteśmy zalogowani na jego konto wystarczy skorzystać z perspektywy `user_tables` w następujący sposób:

```
desc user_tables
```

W wyniku tego polecenia dowiadujemy się, że ma 48 atrybutów, z których jednak potrzebujemy tylko jednego `TABLE_NAME`:

| Nazwa | Wartosc NULL? | Typ |
|-----------------|---------------|---------------|
| TABLE_NAME | NOT NULL | VARCHAR2 (30) |
| TABLESPACE_NAME | | VARCHAR2 (30) |
| CLUSTER_NAME | | VARCHAR2 (30) |
| IOT_NAME | | VARCHAR2 (30) |
| STATUS | | VARCHAR2 (8) |
| PCT_FREE | | NUMBER |
| PCT_USED | | NUMBER |
| INI_TRANS | | NUMBER |
| MAX_TRANS | | NUMBER |
| INITIAL_EXTENT | | NUMBER |
| NEXT_EXTENT | | NUMBER |

| | |
|---------------------------|---------------|
| MIN_EXTENTS | NUMBER |
| MAX_EXTENTS | NUMBER |
| PCT_INCREASE | NUMBER |
| FREELISTS | NUMBER |
| FREELIST_GROUPS | NUMBER |
| LOGGING | VARCHAR2 (3) |
| BACKED_UP | VARCHAR2 (1) |
| NUM_ROWS | NUMBER |
| BLOCKS | NUMBER |
| EMPTY_BLOCKS | NUMBER |
| AVG_SPACE | NUMBER |
| CHAIN_CNT | NUMBER |
| AVG_ROW_LEN | NUMBER |
| AVG_SPACE_FREELIST_BLOCKS | NUMBER |
| NUM_FREELIST_BLOCKS | NUMBER |
| DEGREE | VARCHAR2 (10) |
| INSTANCES | VARCHAR2 (10) |
| CACHE | VARCHAR2 (5) |
| TABLE_LOCK | VARCHAR2 (8) |
| SAMPLE_SIZE | NUMBER |
| LAST_ANALYZED | DATE |
| PARTITIONED | VARCHAR2 (3) |
| IOT_TYPE | VARCHAR2 (12) |
| TEMPORARY | VARCHAR2 (1) |
| SECONDARY | VARCHAR2 (1) |
| NESTED | VARCHAR2 (3) |
| BUFFER_POOL | VARCHAR2 (7) |
| ROW_MOVEMENT | VARCHAR2 (8) |
| GLOBAL_STATS | VARCHAR2 (3) |
| USER_STATS | VARCHAR2 (3) |
| DURATION | VARCHAR2 (15) |
| SKIP_CORRUPT | VARCHAR2 (8) |
| MONITORING | VARCHAR2 (3) |
| CLUSTER_OWNER | VARCHAR2 (30) |

| | |
|--------------|--------------|
| DEPENDENCIES | VARCHAR2 (8) |
| COMPRESSION | VARCHAR2 (8) |
| DROPPED | VARCHAR2 (3) |

Dlatego zadajemy zapytanie ograniczone do jednego atrybutu perspektywy `user_tables`:

```
select table_name from user_tables;
```

i w rezultacie dostajemy potrzebną informację:

```
TABLE_NAME
-----
BONUS
DEPT
EMP
SALGRADE
```

Natomiast chcąc zdobyć tę samą informację, gdy jesteśmy zalogowani jako inny użytkownik niż SCOTT, możemy skorzystać z perspektywy `all_tables`.

Komenda:

```
desc all_tables
```

pokaże nam, że perspektywa `all_tables` ma jeden atrybut więcej niż `user_tables`. Jest to atrybut `owner`, w którym zapisywany jest właściciel tabeli i dlatego jest on nam potrzebny oprócz nazwy tabeli (`table_name`), aby dowiedzieć się przy pomocy perspektyw `all_tables`, jakie tabele posiada użytkownik SCOTT:

```
select table_name, owner from all_tables where owner = 'SCOTT';
```

W rezultacie otrzymujemy te same cztery tabele:

```
TABLE_NAME          OWNER
```



```

-----
BONUS          SCOTT
DEPT           SCOTT
EMP            SCOTT
SALGRADE       SCOTT

```

F. Skrypt tworzący zwykłego użytkownika bazy danych Oracle o schemacie zbliżonym do edukacyjnego schematu użytkownika SCOTT proponowanego przez korporację Oracle do nauki podstaw SQL

```

----- skrypt create_user.sql -----
def USR = 'stud'

def NUM = '0'

CREATE USER &USR&NUM identified by &USR&NUM
DEFAULT tablespace USERS
TEMPORARY tablespace TEMP
-- quota unlimited ON USERS
-- quota unlimited ON TEMP
;

GRANT connect, resource, create view TO &USR&NUM;

-- GRANT UNLIMITED TABLESPACE TO &USR&NUM;

-- DROP TABLE &USR&NUM..DEPT;
CREATE TABLE &USR&NUM..DEPT
  (DEPTNO NUMBER(2),

```

```

    DNAME VARCHAR2(14),
    LOC VARCHAR2(13)
);

-- DROP TABLE &USR&NUM..EMP;
CREATE TABLE &USR&NUM..EMP
(EMPNO NUMBER(4),
ENAME VARCHAR2(10),
JOB VARCHAR2(9),
MGR NUMBER(4),
HIREDATE DATE,
SAL NUMBER(7,2),
COMM NUMBER(7,2),
DEPTNO NUMBER(2)
);

ALTER TABLE &USR&NUM..emp ADD (CONSTRAINT PK_&USR&NUM._EMP_EMPNO
PRIMARY KEY (empno));

ALTER TABLE &USR&NUM..dept ADD (CONSTRAINT
PK_&USR&NUM._DEPT_DEPTNO PRIMARY KEY (deptno));

ALTER TABLE &USR&NUM..emp ADD (CONSTRAINT
FK_&USR&NUM._EMP_DEPTNO_DEPT FOREIGN KEY (deptno) REFERENCES
&USR&NUM..dept (deptno));

-- ALTER TABLE &USR&NUM..emp ADD (CONSTRAINT
-- FK_&USR&NUM._EMP_MGR_EMP FOREIGN KEY (mgr) REFERENCES
-- &USR&NUM..emp (empno));

INSERT INTO &USR&NUM..DEPT VALUES (10,'ACCOUNTING','NEW YORK');
INSERT INTO &USR&NUM..DEPT VALUES (20,'RESEARCH','DALLAS');
INSERT INTO &USR&NUM..DEPT VALUES (30,'SALES','CHICAGO');
INSERT INTO &USR&NUM..DEPT VALUES (40,'OPERATIONS','BOSTON');

```

```

INSERT INTO &USR&NUM..EMP VALUES
(7369, 'SMITH', 'CLERK', 7902, to_date('17-12-1980', 'dd-mm-
      yyyy'), 800, NULL, 20);
INSERT INTO &USR&NUM..EMP VALUES
(7499, 'ALLEN', 'SALESMAN', 7698, to_date('20-2-1981', 'dd-mm-
      yyyy'), 1600, 300, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7521, 'WARD', 'SALESMAN', 7698, to_date('22-2-1981', 'dd-mm-
      yyyy'), 1250, 500, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7566, 'JONES', 'MANAGER', 7839, to_date('2-4-1981', 'dd-mm-
      yyyy'), 2975, NULL, 20);
INSERT INTO &USR&NUM..EMP VALUES
(7654, 'MARTIN', 'SALESMAN', 7698, to_date('28-9-1981', 'dd-mm-
      yyyy'), 1250, 1400, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7698, 'BLAKE', 'MANAGER', 7839, to_date('1-5-1981', 'dd-mm-
      yyyy'), 2850, NULL, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7782, 'CLARK', 'MANAGER', 7839, to_date('9-6-1981', 'dd-mm-
      yyyy'), 2450, NULL, 10);
INSERT INTO &USR&NUM..EMP VALUES
(7788, 'SCOTT', 'ANALYST', 7566, to_date('13-7-1987', 'dd-mm-yyyy') -
      85, 3000, NULL, 20);
INSERT INTO &USR&NUM..EMP VALUES
(7839, 'KING', 'PRESIDENT', NULL, to_date('17-11-1981', 'dd-mm-
      yyyy'), 5000, NULL, 10);
INSERT INTO &USR&NUM..EMP VALUES
(7844, 'TURNER', 'SALESMAN', 7698, to_date('8-9-1981', 'dd-mm-
      yyyy'), 1500, 0, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7876, 'ADAMS', 'CLERK', 7788, to_date('13-7-1987', 'dd-mm-yyyy') -
      51, 1100, NULL, 20);

```

```

INSERT INTO &USR&NUM..EMP VALUES
(7900, 'JAMES', 'CLERK', 7698, to_date('3-12-1981', 'dd-mm-
      yyyy'), 950, NULL, 30);
INSERT INTO &USR&NUM..EMP VALUES
(7902, 'FORD', 'ANALYST', 7566, to_date('3-12-1981', 'dd-mm-
      yyyy'), 3000, NULL, 20);
INSERT INTO &USR&NUM..EMP VALUES
(7934, 'MILLER', 'CLERK', 7782, to_date('23-1-1982', 'dd-mm-
      yyyy'), 1300, NULL, 10);

-- DROP TABLE &USR&NUM..SALGRADE;
CREATE TABLE &USR&NUM..SALGRADE
( GRADE NUMBER,
  LOSAL NUMBER,
  HISAL NUMBER
);

INSERT INTO &USR&NUM..SALGRADE VALUES (1, 700, 1200);
INSERT INTO &USR&NUM..SALGRADE VALUES (2, 1201, 1400);
INSERT INTO &USR&NUM..SALGRADE VALUES (3, 1401, 2000);
INSERT INTO &USR&NUM..SALGRADE VALUES (4, 2001, 3000);
INSERT INTO &USR&NUM..SALGRADE VALUES (5, 3001, 9999);
COMMIT;

undef USR

undef NUM

----- koniec skryptu create_user.sql -----

```