

IBM DB2 a GNU/Linux Debian i zastosowania aplikacyjne

WIELKA MOC

Prezentujemy sposób na instalację IBM DB2 w systemie Debian GNU/Linux.

Przykłady wykorzystania bazy DB2 w połączeniu z językami Java i PHP.

Jeśli jesteś ciekaw, jak uruchomić swoją aplikację z bazą DB2

w środowisku Linux – zapraszam do lektury.

KRZYSZTOF PAŻ

We wcześniejszych numerach magazynu mogli Państwo zapoznać się z bazą danych IBM DB2 dla systemów operacyjnych Linux. Opisaliśmy podstawowe cechy tej platformy bazodanowej, typowe procesy jej instalacji i konfiguracji. W tym artykule kolej na omówienie bardziej profilowanych obszarów praktycznego wykorzystania DB2. Edycje DB2 w wersji 8.2 są wspierane na szereg 32- i 64-bitowych dystrybucji Linuksa – (aktualne szczegóły dostępne pod adresem: <http://www-306.ibm.com/software/data/db2/linux/validate/>) – w szczególności są to głównie dystrybucje typu Suse SLES, Red Hat RHEL, Novell Open Ent. Server, itp. – dedykowane do zastosowań serwerowych. Nie można jednak zapominać, iż mimo dużej popularności serwerowych dystrybucji Red Hat-a za oceanem, a także podobnie Suse w Europie – szczególnie zachodniej, w naszym kraju ugruntowaną pozycję posiadają serwerowe instalacje Linuksa bazujące na innych dystrybucjach – jak choćby Slackware, Mandrake, Debian czy PLD. Pomimo że w oficjalnych dokumentach IBM na temat serwera baz danych DB2 nie wymienia się niektórych dystrybucji Linuksa jako certyfikowane, zalecane i wspierane, nie oznacza to bynajmniej, iż DB2 nie będzie tam działać. Wręcz przeciwnie, można założyć, że użytkownik z powodzeniem zainstaluje i będzie używać DB2 na większości popularnych dys-

trybucji. Zważywszy na pewną specyfikę, a zarazem wysoką jakość i popularność w Polsce dystrybucji Debian GNU/Linux, zajmę się dzisiaj sprawą instalacji serwera IBM DB2 v.8.2 oraz Centrum Informacyjnego DB2 w środowisku dystrybucji GNU/Linux Debian 3.0 Woody (działającej także w przypadku Debian 3.1 Sarge).

Pozyskanie i przygotowanie instalatora DB2

Na początek zaopatrmy się instalator DB2 – na przykład testową 90-dniową wersję enterprise pobieramy ze stron: <https://www106.ibm.com/developerworks/db2/downloads/db2udb-trial/> – naturalnie, jeśli posiadamy jakąś licencjonowaną edycję na nośnikach CD/DVD – dużo wygodniej jest jej użyć. Opisany sposób działa również dla pozostałych edycji DB2 – w szczególności również dla dostępnej bezpłatnie wersji Personal Developers Edition.

Następnie – zależnie od posiadanego formatu instalatora – rozpakowujemy odpowiednio plik archiwum instalacyjnego (np.: `tar zxvf DB2.UDB.ESE.8.2.Linux.tgz`) i przechodzimy do utworzonego w ten sposób katalogu instalatora: `cd 009_ESE_LNX_32_NLV`. Ponieważ zamierzam zaprezentować ogólną metodę instalacji DB2, przydatną do instalacji na dowolnym serwerze – nawet bez środowiska graficznego – użyjemy tekstowego in-

stalatora DB2. Używając dowolnego edytora tekstowego, otwieramy do edycji skrypt instalacyjny – np.: `joe -asis db2/linux/db2_install`. Przechodzimy do sekcji oznaczonej `#Install packages` – okolicie linii 927, a następnie – kilka linii poniżej (około linii 937) odszukujemy linię odpowiadającą za wykonywanie instalacji pakietów DB2 z użyciem mechanizmu rpm:

```
(rpm -ivh ${INSTDEV?}/$  
{pkg?} ${...itd.
```

i zamieniamy jej początek, tak aby program RPM ignorował zależności w instalowanych pakietach rpm – poprzez wstawienie klauzuli `-nodeps`:

```
(rpm -ivh --nodeps ${INSTDEV?}..itd.
```

Następnie zapisujemy tak zmieniony plik i zamykamy edytor.

Przygotowanie naszego systemu Debian GNU/Linux

Od tej pory warto przejść do pracy w trybie administratora: `su -`. Ponieważ instalacja tekstowa bazuje na pakietach rpm – wyposażamy nasz system w adekwatne oprogramowanie RPM Package Manager – poleceniem `apt-get install rpm`, a następnie inicjalizujemy ba-

```

root@db2:~# cd /tmp; baseDB2.v.8.2.INSTALL/009_ESE_LNX_32_NLV
~/baseDB2.v.8.2.INSTALL/009_ESE_LNX_32_NLV ./db2_install

Specify one or more of the following keywords,
separated by spaces, to install DB2 products.

Keyword      Product Description
DB2.ESE      DB2 Enterprise Server Edition for LINUX
DB2.ADCCT   DB2 Administration Client for LINUX
DB2.ADCI    DB2 Application Development Client for LINUX

Enter "help" to redisplay product names.
Enter "quit" to exit.
*****
DB2.ESE

```

Rysunek 1: Instalacja w trybie tekstowym.

zę pakietów rpm: `mkdir /var/lib/rpm` oraz `rpm --initdb`. Ponieważ DB2 może wykorzystywać różnych użytkowników do uruchamiania poszczególnych funkcji serwera, utworzymy odpowiednie konta – odpowiednio dla instancji DB2 (db2) – `adduser db2`, następnie konto wykorzystywane do uruchamiania procedur składowanych – `adduser db2fence` oraz wreszcie trzecie konto dla usługi serwera administracyjnego DB2 – `adduser db2admin`. Naturalnie podane tutaj nazwy są wyłącznie przykładowe – służą one ilustracji treści i ustaleniu uwagi na opisywanych czynnościach – w Państwa systemach produkcyjnych mogą być to dowolne inne, nowo tworzone lub też już uprzednio istniejące nazwy kont.

Instalacja DB2

Po tych przygotowaniach jesteśmy w zasadzie gotowi do rozpoczęcia właściwego procesu instalacji, w tym celu przechodzimy do katalogu zawierającego rozpakowany i przygotowany uprzednio instalator: `cd /ściezka/009_ESE_LNX_32_NLV`. Wydajemy polecenie: `./db2_install` – pojawi się krótka informacja o produktach dostępnych w danym pakiecie instalatora i oczekiwanie na podanie symbolu składnika do instalacji.

Zgodnie z ilustracją – w tym przykładzie – aby zainstalować wersję enterprise, wprowadzamy symbol: `DB2.ESE` i potwierdzamy klawiszem Enter. Następnie oczekujemy na zakończenie procesu instalacji. Jeśli pragniemy wykorzystywać procedury składowane w języku Java lub też dany komputer ma konsolę graficzną i zamierzamy wykorzystywać na nim lokalnie graficzne narzędzia administracyjne, warto jeszcze zainstalować biblioteki środowiska Java w wersji IBM – poleceniem: `rpm -i --nodeps db2/linux/java-1.4/IBMJava2-SDK-1.4.1-2.0.i386.rpm`. Po zakończeniu tej operacji możemy sprawdzić listę pakietów typu RPM, które zainstalowały się w naszym systemie: `rpm -qa`.

Następnie warto utworzyć instancję serwera administracyjnego DB2, który umożliwi nam zdalną, zunifikowaną administra-

cję za pomocą graficznych narzędzi administracyjnych DB2. W tym celu wydajemy polecenie:

```

/opt/IBM/db2/V8.1/instance/dasrcrt db2admin

```

Równie lub nawet bardziej użyteczną czynnością będzie utworzenie instancji dla baz DB2 – `/opt/IBM/db2/V8.1/instance/db2icrt -a SERVER -`

`p DB2_db2 -s ese -u db2fence db2` – jak widać, składnia tego polecenia zawiera nazwy utworzonych wcześniej kont – na przykład db2, które będzie właścicielem danej instancji. W tym miejscu warto jeszcze wspomnieć, iż pomocną opcją jest także możliwość automatycznego startowania wybranych instancji podczas uruchomienia systemu serwera. Jeśli np. chcielibyśmy, aby to właśnie instancja o nazwie „db2” była uruchamiana przy starcie systemu, należy wydać polecenie: `/opt/IBM/db2/V8.1/bin/db2iauto -on db2`

Jeśli wszystkie opisane kroki się powiodły, możemy uznać, że podstawowy etap instalacji serwera DB2 został zakończony – jednak zwykle warto to zweryfikować w praktyce. W tym celu przechodzimy na konto właściciela utworzonej instancji baz DB2: `su - db2` i sprawdzamy swoje środowisko – poleceniem `set`. Jeżeli w naszej ścieżce PATH nie odnajdujemy katalogów z cyklu: `/home/db2/sqllib/[bin, misc, adm]` lub brak jest ustawionych zmiennych typu `DB2INSTANCE` lub `INSTHOME` – to oznacza, że powinniśmy uzupełnić naszą konfigurację środowiska – w zależności od używanej powłoki dla konta db2 – dokonujemy tego w odpowiednich plikach konfiguracyjnych powłoki. W przypadku bash'a warto na końcu pliku „.bash_profile” dodać linię:

```

. /home/db2/sqllib/db2profile

```

Dzięki temu nasza powłoka logowania na koncie db2 powinna mieć ustawione właściwe środowisko pracy. Domyślnie, linia konfiguracji środowiska db2 powinna zostać dodana podczas tworzenia instancji do pliku `„/home/db2/.bashrc”` – co ustawia środowisko dla powłok wywoływanych w inny sposób niż przy logowaniu. Sprawdźmy, czy tak się stało w istocie. Wylogujemy się z konta db2 `exit` i zalogujemy na nie ponownie `su - db2`. Wydajmy teraz polecenie uruchamiające bieżącą instancję bazy danych: `db2start` – po chwili powinien się pojawić komunikat `SQL1063N` informujący, że polecenie się po-

wiodło, zaś podstawowe procesy bieżącej instancji działają z prawami użytkownika db2 – co, oczywiście, łatwo zweryfikujemy poleceniem `ps aux`.

Dla pewności spróbujmy jeszcze, jako użytkownik db2, utworzyć przykładową, testową bazę danych poleceniem: `db2 create db baza01`. Po chwili, przeznaczonej na alokowanie przez serwer domyślnie przyjętych wartości dla tworzonej bazy, powinniśmy otrzymać komunikat `DB20000I`, niosący informację, że tworzenie bazy się powiodło. Możemy zatem

```

root@db2:~# cd /tmp; baseDB2.v.8.2.INSTALL/009_ESE_LNX_32_NLV
~/baseDB2.v.8.2.INSTALL/009_ESE_LNX_32_NLV ./db2_install

Specify one or more of the following keywords,
separated by spaces, to install DB2 products.

Keyword      Product Description
DB2.ESE      DB2 Enterprise Server Edition for LINUX
DB2.ADCCT   DB2 Administration Client for LINUX
DB2.ADCI    DB2 Application Development Client for LINUX

Enter "help" to redisplay product names.
Enter "quit" to exit.
*****
DB2.ESE

```

Rysunek 2: Podłączenie do bazy.

podłączyć się do nowej bazy `db2 connect to baza01` i rozpocząć pracę.

Zgodnie z ilustracją – otrzymujemy możliwość pracy z nową bazą danych.

Do zatrzymywania pracy instancji służy polecenie `db2stop` – wydajmy zatem taką komendę, a następnie przekonamy się, iż serwer odmówił nam zatrzymania instancji ze względu na aktywne połączenia. Dokładniej, chodzi o nasze własne polecenie „db2 connect...” wydane wcześniej – bowiem klient tekstowy db2, działający w sesji naszego terminala pamięta to nasze połączenie jako aktywne. Będzie tak, dopóki nie zamkniemy bieżącej sesji powłoki, w której wydaliśmy to polecenie lub też nakażemy `explicit` go zakończyć komendą: `db2 disconnect baza01` – następnie, możemy już zatrzymać instancję poleceniem `db2stop`.

Myszę, że w tym miejscu warto wykonać pełny restart systemu Linux i sprawdzić, czy instancja db2 zostanie automatycznie uruchomiona przy starcie systemu.

Centrum Informacyjne DB2

Z dystrybucją IBM DB2 dostarczane jest Centrum Informacyjne DB2, które stanowi doskonałą dokumentację serwera bazodanowego i towarzyszących mu narzędzi. Naturalnie dostępne jest ono także w wersji online w Internecie <http://publib.boulder.ibm.com/infocenter/db2help/>. Jeśli jednak ze względu na prędkość łączy czy też po prostu komfort pracy offline chcielibyśmy je mieć na naszym serwerze z systemem Debian GNU/Linux – jest to, naturalnie, możliwe. Jeżeli zapodzialiśmy gdzieś płytę instalacyjną z Information Center lub też nie jest ona już w najnowszej wersji – możemy spróbować odwiedzić adres: <ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/db2docv8/UNIX>

/FP7_U498354/, znajduje się tam właśnie IBM DB2 Information Center do wersji 8.2 w postaci pliku FP7_U498354_DOC_LINUX.tar. Plik ten rozpakowujemy i przechodzimy do utworzonego w ten sposób katalogu z instalatorem `cd 261_DOCE_LNX_3264_NLV`. Następnie, podobnie jak w przypadku instalacji serwera DB2, modyfikujemy w skrypcie `joe db2/linux/db2_install` linię 573 w sekcji „# Install packages” odpowiadającą za instalację pakietów rpm – tak, aby ignorowała zależności między pakietami (`rpm -ivh --nodeps... itd.`). Dalej zaś, najlepiej z prawami administratora, wywołujemy znane już polecenie instalacji w trybie tekstowym `db2_install` i jako nazwę obiektu do instalacji podajemy nazwę modułu dokumentacji „DB2.DOCE” – w polu oczekiwania skryptu tekstowego. Po zakończeniu skryptu dokumentacja znajdzie się w odpowiednich katalogach, zaś uruchomienie modułu usługi Information Center jest możliwe poprzez wywołanie skryptu startowego `/etc/init.d/db2icd start`. Od tej pory (po uruchomieniu przez `db2icd`) możliwe jest odwołanie do naszej lokalnej kopii dokumentacji z Centrum Informacyjnego poprzez otwarcie w przeglądarce adresu: <http://moj.serwer.com.pl:51000>, oczywiście, w przypadku częstego bądź ciągłego wykorzystania warto być może dodać automatyczne startowanie i zatrzymywanie usługi centrum informacyjnego poprzez odpowiednie ręczne ustawienie linków symbolicznych w strukturze katalogów `/etc/rcX.d/` albo posłużenie się poleceniem: `update-rc.d db2icd defaults 99` – co utworzy automatycznie strukturę linków.

Java a DB2

Wykorzystanie IBM DB2 jako repozytorium danych dla aplikacji tworzonych w języku Java jest bardzo proste i intuicyjnie podobne do innych relacyjnych baz danych. Założmy, że chcemy używać sterownika JDBC typu 4 do komunikacji z serwerem DB2. W tym celu rozpoczniemy od zlokalizowania biblioteki z tym sterownikiem. Domyślnie znajduje się ona w następującym miejscu: `/opt/IBM/db2/V8.1/java/db2jcc.jar`. Przy kompilacji lub uruchamianiu programów korzystających z połączenia JDBC do DB2 warto dodać ten plik „db2jcc.jar” do ścieżki CLASSPATH. Pozostałe zagadnienia programistyczne są bardzo zbliżone do innych typowych sytuacji użycia JDBC. Między innymi mamy do dyspozycji następujące możliwości:

String połączeniowy możemy zadeklaro-

wać jako:

```
String urlPrefix = „jdbc:db2://  
localhost:60000/baza01”;
```

Ładowanie klasy realizującej obsługę JDBC type 4 z dla DB2 realizujemy:

```
Class.forName  
("com.ibm.db2.jcc.DB2Driver");
```

Jeśli chcemy nawiązać połączenie z zadeklarowaną w „urlPrefix” bazą, wykonujemy:

```
con = DriverManager.getConnection(  
urlPrefix, "user", "haslo");
```

Dalej jest już zupełnie klasycznie – czyli tworzymy „statement”:

```
stmt = con.createStatement();
```

Ustalamy treść zapytania:

```
String SQLquery = "select *  
from customers”;
```

Wykonujemy zapytanie:

```
rs = stmt.executeQuery(SQLquery);
```

Teraz pozostało nam już tylko odebrać i zaprezentować użytkownikowi wyniki, a następnie posprzątać nieco po sobie – najlepiej zamykając resultset:

```
rs.close();
```

a także statement:

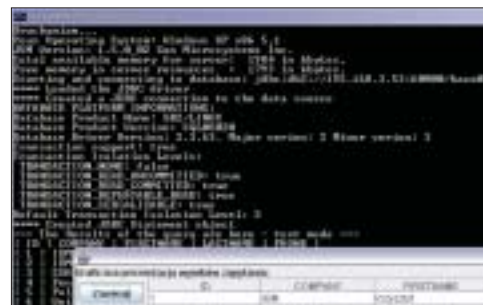
```
stmt.close();
```

oraz w końcu samo połączenie JDBC:

```
con.close();
```

W ten sposób możemy wykonać aplikację korzystającą z DB2 zainstalowanego na systemie Debian GNU/Linux za pomocą JDBC z dowolnej platformy klienckiej.

Jak zapewne Państwo zauważyli, na ilustracji obserwujemy podłączenie i rezultaty działania aplikacji Java w systemie Windows XP w oparciu o serwer IBM DB2 8.2 działający pod kontrolą Linuksa – a dokładnie dystrybucji Debian.



Rysunek 3: Połączenie JDBC do bazy DB2.

Używamy DB2 z poziomu PHP

Domyślnie dostarczana dystrybucja binarna PHP4 w systemie Debian GNU/Linux nie posiada prekompilowanej opcji umożliwiającej wykorzystanie serwera DB2 wprost z PHP. Dlatego też należy pobrać najnowszą wersję źródłową PHP – na przykład ze strony: <http://www.php.net/downloads.php#v4> – obecnie jest to 4.3.11, rozpakować archiwum i dokonać konfiguracji pożądanych opcji oraz kompilacji i instalacji w systemie. Nim jednak przejdziemy do szczegółów, należy wspomnieć, iż w systemie Debian GNU/Linux do kompilacji PHP potrzebnych jest szereg bibliotek, dlatego też zainstalujemy je wcześniej:

```
apt-get install apache-dev libxml2  
xml-core libxml2-dev libxml2-utils  
unixODBC libct1 unixODBC-dev  
libbz2-dev libjpeg-dev libpng-dev  
libcurl-dev libxpm-dev libfreetype  
pe6-dev freetype2-dev libt1-dev  
freetype1-tools t1utils libgd-dev  
libpam0g-dev libc-client-dev  
uw-mailutils libldap2-dev libm  
crypt-dev libmhash-dev libpcre3-dev  
librecode-dev libsnmp5-dev  
freetds-dev mcrypt bison flex m4  
libzzip-dev
```

Teraz (pamiętając o przyjętym w niniejszym tekście założeniu, iż właścicielem instancji jest użytkownik `db2` posiadający swoje konto w katalogu `/home/db2`) możemy przejść do katalogu z rozpakowanymi źródłami PHP4 i wydać polecenie konfiguracji środowiska do kompilacji z naszymi opcjami:

```
./configure --prefix=/usr --with-  
-apxs=/usr/bin/apxs --with-ibm  
-db2=/home/db2/sqllib --syscon  
fdir=/etc --with-config-file  
-path=/etc/php4/apache --with-uni  
xODBC=shared,/usr --enable-dbx
```



Rysunek 4: Wynik kompilacji PHP ze wsparciem dla DB2.

Jeśli w naszym systemie obecny był już wcześniej PHP4, systemowy plik *php.ini* możemy pozostawić bez zmian, jeśli jednak mieliśmy apache bez PHP, warto teraz skopiować plik *php.ini-recommended* z katalogu ze źródłami PHP do wyspecyfikowanej lokalizacji */etc/php4/apache/php.ini* – w tym przykładzie. Po rozważeniu tego zagadnienia jesteśmy gotowi do kompilacji. Wydajemy zatem polecenie: *make*, a następnie *make install* oraz dla pewności wszelkiej: */sbin/ldconfig*. Warto teraz sprawdzić, czy w pliku „*/etc/apache/httpd.conf*” znajdują się odpowiednie wpisy (powinny się tam pojawić wskutek działania polecenia „*make install*”) odpowiadające za do-dawanie modułu php:

```
AddModule mod_php4.c
```

ładowanie modułu php4:

```
LoadModule php4_module \
/usr/lib/apache/1.3/libphp4.so
```

i obsługę aplikacji php:

```
AddType application/x-httpd-php \
.php
```

w serwerze apache. Jeśli tak właśnie jest, możemy więc utworzyć gdzieś w katalogach naszych stron www plik *phpinfo.php* o znajomej treści:

```
<? phpinfo(); ?>
```

Nadszedł już czas na test naszej działalności. W pliku */etc/init.d/apache* należy zapewnić dostępność do tych samych zmiennych środowiskowych, które posiada w powłoce użytkownik db2 - czyli znanym już sposobem, przed liniami uruchamiającym

demony httpd dodajemy linię konfiguracji środowiska:

```
./home/db2/sql/lib/db2profile
```

i wreszcie restartujemy serwer apache: */etc/init.d/apache restart*, następnie zaś wskazujemy w przeglądarce adres pliku testowego php <http://192.168.3.53/www-db2/phpinfo.php> i powinniśmy oglądać efekt naszego działania - podobny nieco do następnego ilustracji.

Kolejne wiersze w tabelach informacyjnych systemu umożliwiają weryfikację ustawień nowej, działającej wersji PHP. Czas zatem na praktyczne wykorzystanie bazy w skrypcie PHP – dokonamy tego za pomocą interfejsu CLI DB2 użytkowanego z poziomu PHP jako połączenie w standardzie ODBC. Do bazy na tym samym serwerze łączymy się klasycznie:

```
$dbconn = odbc_connect($dbname, \
$username, $password);
```

gdzie zmienna *\$dbname* oznacza nazwę bazy – np.: „baza01”, zaś w przypadku połączenie do bazy na innej maszynie niż środowisko wykonywania skryptów, można użyć następującej konstrukcji:

```
$dbname = "DRIVER={IBM DB2 ODBC \
DRIVER};HOSTNAME=db2serwer.domena.\
com.pl;DATABASE=SAMPLE;PROTO\
```

```
COL=TCPIP; PORT=1527;"
```

i dalej już postępujemy według poprzedniego schematu. Następnie ustalamy treść zapytania do bazy danych - np.:

```
$select_stmt = "SELECT last_name, \
first_name, middle_initial, au\
thor_id FROM author ";
```

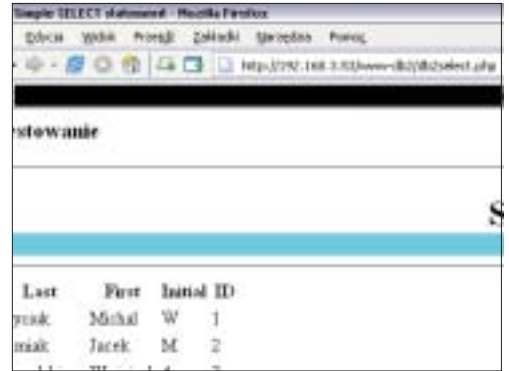
i w sposób klasyczny wykonujemy zapytanie:

```
$result = odbc_exec($dbconn, $se\
lect_stmt);
```

W przypadku niepowodzenia możemy od-czytać komunikat błędu poprzez instrukcję:

```
$sqlerror = odbc_errormsg($dbconn);
```

Ponieważ, naturalnie wszystko działa, pobieramy i prezentujemy wyniki - na przykład



Rysunek 5: Proste zapytanie SQL z PHP na tabeli w bazie IBM DB2.

w pętli while:

```
while ($row = odbc_fetch_array(\
$result)) {...}
```

W podobny sposób możemy używać pozostałych poleceń języka SQL z poziomu PHP.

W tym krótkim artykule nie sposób opisać wszystkich aspektów związanych z wykorzystaniem aplikacyjnym IBM DB2, jednakże mam nadzieję, że przedstawione tu informacje pomogą Państwu rozpocząć przygodę z praktycznym użytkowaniem IBM DB2 na systemie GNU/Linux – w szczególności, jeśli jest to dystrybucja Debian. Na zakończenie zamieszczam kilka przydanych odnośników do ciekawych miejsc w Sieci, które mogą być pomocne w tego typu sprawach. ■

AUTOR

Krzysztof Paż. Absolwent Instytutu Informatyki na Wydziale Elektroniki i Technik Informatycznych Politechniki Warszawskiej. Obecnie zajmuje się projektowaniem i sprzedażą rozwiązań z dziedziny zarządzania informacją w oparciu o produkty i technologie IBM Information Management Software - w tym również DB2.

INFO

[1] <http://www-128.ibm.com/developerworks/db2/newto/new-to-db2udb.html>

[2] <http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0502scott/>

[3] <http://www-106.ibm.com/developerworks/linux/edu/l-dw-linuxperl-i.html>

[4] <http://www.ibm.com/developerworks/db2>

[5] <http://www.ibm.com/db2/linux/>

Dodatkowe materiały do artykułu:
<http://linux-magazine.pl/issue/19/db2.tgz>