

Paweł DRZYMAŁA, Henryk WELFLE
Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych

MECHANIZMY WSPARCIA URUCHAMIANIA APLIKACJI ORACLE W ŚRODOWISKU WERSJI IBM DB2 9.7.5 EXPRESS-C, PRZEZNACZONEJ DLA SPOŁECZNOŚCI

Streszczenie. W artykule przedstawiono mechanizmy wsparcia uruchamiania aplikacji Oracle w środowisku wersji DB2 9.7.5 Express-C. Zaprezentowano możliwości kompilacji i wykonywania aplikacji w DB2 9.7 Express-C z użyciem języków SQL i PL/SQL oraz obiektów Oracle. W wersji tej wprowadzono wsparcie Oracle 10g i 11g, które dostępne było wcześniej tylko dla komercyjnej edycji Enterprise. W odniesieniu do starszych komercyjnych wydań DB2, wprowadzone zostały nowe poziomy kompatybilności dla systemu Oracle.

Słowa kluczowe: IBM DB2 9.7.5 Express-C, Oracle, Kompatybilność Oracle w DB2, DB2_COMPATIBILITY_VECTOR

MECHANISMS OF SUPPORT FOR ORACLE APPLICATIONS IN VERSION 9.7.5 IBM DB2 EXPRESS-C ENVIRONMENT INTENDED FOR COMMUNITY

Summary. The article presents the mechanisms of support to run Oracle applications in DB2 9.7.5 Express-C environment. It presents the possibility of compilation and execution applications in the DBMS DB2 9.7 Express-C using SQL, PL/SQL and Oracle objects. This version implements support of Oracle 10g and 11g, which was previously available only in Enterprise Edition. In relation to older commercial versions it introduced new levels of Oracle compatibility. The article presents also how to configure DB2 to work with the PL/SQL language.

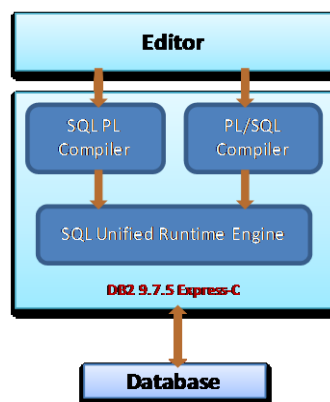
Keywords: IBM DB2 9.7.5 Express-C, Oracle, Compatibility Oracle in DB2, DB2_COMPATIBILITY_VECTOR

1. Wprowadzenie.

System zarządzania bazami danych IBM DB2 Express-C (Community) jest produktem darmowym, o bardzo dużych możliwościach, przeznaczonym do zarządzania danymi składowanymi w klasycznych, relacyjnych strukturach oraz w postaci struktury hierarchicznej XML. Nie jest jednak udostępniany na zasadach open-source. DB2 9.7 Express-C pozwala na obsługę dialektów języków Oracle SQL i PL/SQL (dla wersji Oracle 10g i 11g) od wersji DB2 9.7.5. Dzięki temu wiele aplikacji napisanych dla systemu zarządzania bazami danych Oracle ze zmianami może zostać uruchomionych w środowisku DB2 (podaje się szacunkowo, że około 98% składni języka PL/SQL i około 90% obiektów PL/SQL może być bezpośrednio transferowana do DB2 – orientacyjne wartości liczbowe zostały podane na podstawie przykładowego raportu wygenerowanego narzędziem MEET DB2). Należy podkreślić, że pomimo możliwości przenoszenia kodu PL/SQL do systemu DB2, nie zostały zaimplementowane wszystkie rozszerzenia, np. obiektowo-relacyjnego programowania (np. NESTED TABLE). Jest to istotny element wpływający na możliwości realizacji migracji danych i aplikacji między systemami. W wersjach komercyjnych licencji DB2, instalowanych na systemach Linux, Unix i Windows (DB2 LUW), można federować dane z innymi bazami DB2 LUW. W przypadku wykonywania zapytań rozproszonych w stosunku do systemu Oracle, należy doinstalować opakowania relacyjne dla tego systemu, wraz z licencją. Szczegóły tego rozwiązania przedstawiono w artykule [1].

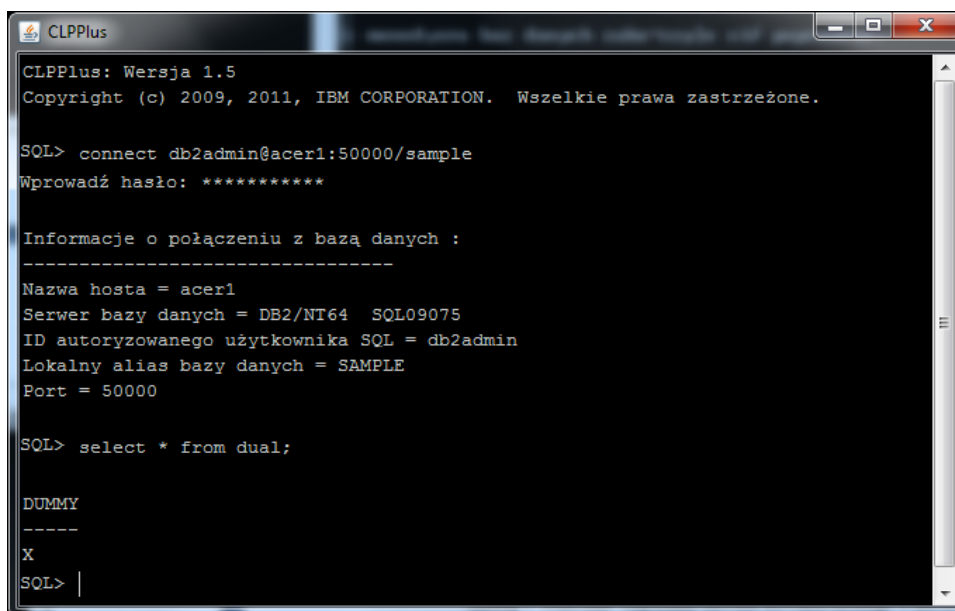
Użytkownicy, którzy pracują z bazą danych Oracle, mogą w wersji DB2 9.7.5 Express-C skorzystać z wielu nowych poziomów kompatybilności. Modyfikacje te ułatwiają łatwe włączenie aplikacji Oracle w celu ich szybkiego uruchamiania w środowiskach DB2.

DB2 może obsługiwać składnię języka PL/SQL, oprócz własnego dialektu języka SQL PL. Sposób wsparcia dla języka PL/SQL oraz SQL PL przedstawiono na rys. 1.



Rys. 1. Wsparcie dla języka PL/SQL
Fig. 1. Support for the language PL/SQL

W celu zapewnienia kompatybilności z systemem Oracle w wersji DB2 9.7, dodano obsługę narzędzia wiersza polecenia CLPPlus, które pozwala na uruchamianie poleceń SQL oraz PL/SQL i jest podobny do istniejącego procesora wiersza polecenia DB2 (CLP). Na rys. 2. zaprezentowano tekstowe środowisko CLPPlus.



```
CLPPlus: Wersja 1.5
Copyright (c) 2009, 2011, IBM CORPORATION.  Wszelkie prawa zastrzeżone.

SQL> connect db2admin@acer1:50000/sample
Wprowadź hasło: *****

Informacje o połączeniu z bazą danych :
-----
Nazwa hosta = acer1
Serwer bazy danych = DB2/NT64  SQL09075
ID autoryzowanego użytkownika SQL = db2admin
Lokalny alias bazy danych = SAMPLE
Port = 50000

SQL> select * from dual;

DUMMY
-----
X
SQL> |
```

Rys. 2. Narzędzie CLPPlus

Fig. 2. Tool CLPPlus

Wsparcie techniczne dla większości typów danych PL/SQL zostało zaimplementowane standardowo w systemie DB2, a pozostałe można włączać przez zmienną rejestru. Można kontrolować, które z tych zgodności są włączone, za pomocą ustawienia zmiennej rejestru DB2_COMPATIBILITY_VECTOR, opisane w dalszej części artykułu.

DB2 umożliwia tworzenie funkcji zdefiniowanych przez użytkownika, w których można zawrzeć często używane funkcjonalności. Przykładowo, gdy przeprowadza się migrację aplikacji z bazy Oracle do DB2, często wywołuje się funkcję Oracle o nazwie NVL(). Nie istnieje ona w DB2, ale można ją utworzyć z jej funkcjonalnością. Nie jest to jednak sposób docelowy i elegancki, stąd wprowadzono wiele poziomów kompatybilności dla systemu Oracle w DB2.

2. Zachowanie zgodności Oracle w systemie DB2

Zachowanie zgodności Oracle w systemie DB2 realizowane jest przez:

- rozszerzenia języka SQL, typów i dialektu języka PL/SQL,
- procedury i funkcje,
- wbudowane pakiety,

- ulepszoną współbieżność w stosunku do poprzednich wersji DB2,
- zgodność schematu,
- zgodność SQL*Plus.

Większość aplikacji Oracle wykorzystuje wiele wbudowanych pakietów udostępnionych przez serwer, które umożliwiają wykonywanie wielu operacji typu raportowanie, wysyłanie poczty elektronicznej czy zwiększających możliwości komunikacyjne. Aby spełnić warunek kompatybilności przenoszonej aplikacji bez potrzeby pisania wykorzystywanych pakietów, mamy część pakietów zaimplementowanych w DB2. Zaimplementowane pakiety to: DBMS_OUTPUT, UTL_MAIL, UTL_SMTP, UTL_FILE, UTL_DIR, DBMS_UTILITY, DBMS_SQL, DBMS_ALERT, DBMS_JOB, DBMS_LOB, DBMS_PIPE.

Z punktu widzenia programisty Javy pojawiły się nowe funkcjonalności dodane do sterownika JDBC DB2, które są wykorzystywane do wspierania serwera Oracle za pomocą sterownika JDBC Oracle. Wiele starszych aplikacji Oracle C korzysta z Oracle Client Interface (OCI) do komunikacji z systemem bazodanowym. DB2 i zawiera OCI zgodny z klientem Oracle o nazwie DB2CI.

Istnieje w DB2 narzędzie CLPPlus, które jest zgodne z narzędziem SQL*Plus w systemie Oracle, przeznaczone wykonywania skryptów czy pojedynczych poleceń administracyjnych.

W każdym nowo wprowadzanym FixPack-u do systemu DB2 9.7 pojawiają się nowe funkcjonalności związane z kompatybilnością z systemem Oracle.

W celu zwiększenia kompatybilności między systemami ważnym elementem jest odpowiednia kontrola współbieżności, najlepiej na tym samym poziomie. Aktualnie DB2 wykorzystujący nowy poziom izolacji transakcji CS (Cursor Stability z opcją Currently Committed), zachowuje się tak samo, jak transakcje w systemie Oracle, wykorzystujące domyślny poziom izolacji. Przy tym poziomie izolacji mamy następujące zachowania:

- czytający nie blokuje czytającego,
- czytający nie blokuje zapisującego,
- zapisujący nie blokuje czytającego,
- zapisujący blokuje zapisującego.

Mamy także inne poziomy izolacji transakcji w DB2, jak:

- odczyt niezatwierdzonych danych (Uncommitted Read – UR),
- stabilność kursora (Cursor Stability – CS),
- stabilność odczytu (Read Stability – RS),
- odczyt powtarzalny (Repeatable Read – RR).

3. Ustawienie DB2 dla umożliwienia wykonywania aplikacji Oracle

W celu uruchamiania aplikacji Oracle na serwerach DB2 należy odpowiednio go skonfigurować. Służy do tego zmienna rejestru DB2_COMPATIBILITY_VECTOR, ustawiana za pomocą polecenia db2set. Zmienna rejestru DB2_COMPATIBILITY_VECTOR służy do włączania jednej lub więcej cech kompatybilności DB2, które zostały wprowadzone do DB2 9.7.5 Express-C (dla wersji komercyjnych od wersji 9.5). Poziomy kompatybilności są różne w różnych wersjach. Pełny poziom kompatybilności uzyskuje się przez przypisanie zmiennej rejestru DB2_COMPATIBILITY_VECTOR wartości ORA (odpowiednik wartości heksadecymalnej 10FFF). Należy zwrócić uwagę, iż bit 13 i 14 zmiennej rejestru w tym przypadku jest 0.

Ustawienie zmiennej rejestru umożliwiającej obsługiwane wszystkich funkcji zgodności Oracle, przedstawiono na przykładzie:

```
db2set DB2_COMPATIBILITY_VECTOR=ORA
db2stop
db2start
```

Ustawienie zmiennej rejestru wyłączającej obsługę zgodności z Oracle, przedstawiono na przykładzie:

```
db2set DB2_COMPATIBILITY_VECTOR=
db2stop
db2start
```

W zależności od potrzeb mamy możliwość włączenia tylko wybranych funkcjonalności związanych z kompatybilnością z systemem Oracle. Przykładowo, możemy włączyć obsługę nazwy DUAL i operatora outer join (+) przez wykonanie poleceń:

```
db2set DB2_COMPATIBILITY_VECTOR=06
db2stop
db2start
```

W rozdziale czwartym przedstawiono wszystkie dostępne funkcjonalności związane z kompatybilnością z Oracle w DB2 9.7.5 Express-C.

4. Zmienna rejestru DB2_COMPATIBILITY_VECTOR

Zmienna rejestru DB2_COMPATIBILITY_VECTOR może ustawiać poszczególne funkcjonalności dostępne w bazie Oracle (nie wszystkie dotyczą systemu Oracle, część z nich dotyczy systemu Sybase czy SQL Server) w zależności od potrzeb klienta. Należy zwrócić uwagę, iż pewne funkcjonalności (zaznaczone (*) w tabeli 1) mogą zostać włączone, ale nie mogą być wykorzystane w już istniejących bazach danych. Ma to zastosowanie tylko dla nowo utworzonych baz danych. Wartość zmiennej rejestru podaje się jako wartość heksadecy-

malną lub wykorzystuje się wartość ORA dla włączenia wszystkich funkcjonalności. Nowe ustawienie zmiennej rejestru nie zaczną obowiązywać do czasu zatrzymania instancji i jej ponownego uruchomienia.

Tabela 1

Opcje ustawień zmiennej DB2_COMPATIBILITY_VECTOR

Pozycja bitu	Funkcja zgodności z Oracle	Opis
1	ROWNUM	Umożliwia stosowanie pseudokolumny ROWNUM w zapytaniach SQL, w klauzuli WHERE (odniesienie do funkcji analitycznej ROW_NUMBER() OVER()).
2	DUAL	Pozwala na stosowanie tabeli DUAL, wykorzystywanej w Oracle jako odniesienie do tabeli SYSIBM.DUAL.
3	Operator outer join	Włącza możliwość stosowania operatora (+) w złączeniach zewnętrznych.
4	Zapytania hierarchiczne	Umożliwia wykorzystanie klauzuli CONNECT BY w zapytaniach hierarchicznych.
5	Typ danych NUMBER(*)	Włącza możliwość stosowania typu danych NUMBER.
6	Typ danych VARCHAR2 (*)	Włącza możliwość wykorzystania typu danych VARCHAR2.
7	Typ danych DATE (*)	Włącza możliwość wykorzystania typu danych DATE.
8	TRUNCATE TABLE	Włącza semantykę polecenia TRUNCATE.
9	Obsługa łańcuchów znaków	Umożliwia przypisanie typu danych w DB2 CHAR lub GRAPHIC (zamiast typu danych VARCHAR lub VARGRAPHIC) o stałej długości, dla długości bajtu mniejszej lub równej 254. (DB2)
10	Metody dla kolekcji	Włącza możliwość korzystania z metod wykonywania operacji na tablicach.
11	Widoki zgodne ze słownikiem danych (*)	Umożliwia korzystanie z kompatybilnych widoków zgodnych ze słownikami danych występujących w Oracle.
12	Kompilacja PL/SQL	Składnia języka PL/SQL może być kompilowana i wykonywana przy użyciu DB2
13	Insensitive cursor	Umożliwia definiowanie kursorów z opcją WITH RETURN, nieczuły na zmiany bez opcji FOR UPDATE (Sybase i SQL Server)
14	Paramter INOUT	Umożliwia wykorzystanie specyfikacji DEFAULT dla deklaracji parametru INOUT (Sysbase i SQL Server)
17	Poziomy dostęp do danych dla poleceń SQL	Możliwy jest do ustawienia różny poziom dostępu do danych od strony procedur przechowywanych i funkcji (zdefiniowanych przez użytkownika).

4.1. Zgodność związana z ROWNUM

Opcja ta umożliwia stosowanie pseudokolumny ROWNUM w zapytaniach SQL w klauzuli WHERE (odniesienie do funkcji analitycznej ROW_NUMBER() OVER()).

Przykład:

```
SELECT deptname FROM department
WHERE ROWNUM between 11 and 20
ORDER BY 1;
```

Należy zwrócić uwagę na fakt, iż wynik zapytania wykorzystujący pseudokolumnę ROWNUM w klauzuli WHERE inaczej jest interpretowany w Oracle. W DB2 najpierw realizowane jest klauzula sortowana, a następnie klauzula ograniczająca WHERE, wykorzystująca ROWNUM. W systemie Oracle najpierw następuje odwołanie do pseudokolumny ROWNUM w klauzuli WHERE, a następnie zestaw zwróconych krotek poddawany jest operacji sortowania klauzuli ORDER BY. Dodatkowo, w systemie Oracle wynik przykładowego zapytania, wykorzystujący operator BETWEEN AND, nie zwróci żadnego rekordu, jeśli nie zaczniemy wybierać elementy od pierwszego.

4.2. Zgodność związana z tabelą DUAL

Opcja ta pozwala na wykorzystanie tabeli DUAL (jako odniesienie do tabeli DUAL znajdującej się w schemacie SYSIBM w DB2) wykorzystywanej w Oracle.

Przykład:

```
SELECT SYSDATE FROM DUAL;
```

4.3. Zgodność związana z operatorem outer join

Opcja ta włącza możliwość wykorzystania operatora (+) w złączeniach zewnętrznych. W systemie Oracle zamiast złączeń typu LEFT OUTER JOIN i RIGHT OUTER JOIN można wykorzystywać operator (+). Jedna i druga forma jest poprawna.

Przykład:

```
SELECT DEPTNAME, LASTNAME, JOB
FROM DEPARTMENT D LEFT OUTER JOIN EMPLOYEE E
ON D.DEPTNO=E.WORKDEPT;
lub
SELECT DEPTNAME, LASTNAME, JOB
FROM DEPARTMENT D, EMPLOYEE E
WHERE D.DEPTNO=E.WORKDEPT (+);
```

4.4. Zapytania hierarchiczne

Opcja ta umożliwia wykorzystanie klauzuli CONNECT BY w zapytaniach hierarchicznych.

Przykład:

```
SELECT ENAME, LEVEL, SAL,
CONNECT_BY_ROOT ENAME AS ROOT,
SUBSTR(SYS_CONNECT_BY_PATH(ENAME, ':'), 1, 20) HIERARCHIA
FROM EMP
START WITH ENAME = 'KING'
CONNECT BY PRIOR EMPNO = MGRNO
ORDER SIBLINGS BY SAL;
```

4.5. Typ danych NUMBER

Opcja ta włącza możliwość wykorzystania typu danych NUMBER.

Przykład:

```
CREATE TABLE TEST (ID NUMBER(5,2));
```

4.6. Typ danych VARCHAR2

Opcja ta włącza możliwość wykorzystania typu danych VARCHAR2 i NVARCHAR2.

Przykład:

```
CREATE TABLE TEST (NAME VARCHAR2(15));
```

4.7. Typ danych DATE

Opcja ta włącza możliwość wykorzystania typu danych DATE, który oznacza w DB2 typ TIMESTAMP(0). Ten typ danych przechowuje informacje związane z datą i czasem.

4.8. TRUNCATE TABLE

Opcja ta umożliwia skasowanie wszystkich danych w danej tabeli, bez możliwości wycofania tej operacji w celu ponownego wykorzystania przydzielonego miejsca.

Przykład przedstawia czyszczenie danych z tabeli EMP, niezależnie od ustawionych wyzwalaczy, z zachowaniem miejsca przeznaczonego na późniejsze użycie:

```
TRUNCATE TABLE EMP  
  REUSE STORAGE  
  IGNORE DELETE TRIGGERS  
  IMMEDIATE;
```

4.9. Obsługa łańcuchów znaków (dotyczy tylko DB2)

Opcja ta umożliwia przypisanie typu danych CHAR lub GRAPHIC w DB2 (zamiast typu danych VARCHAR lub VARGRAPHIC dla długości bajtu ≥ 255) o stałej długości, którego długość bajtu jest mniejsza lub równa 254, w celu poprawności wykonywania zapytań SQL.

4.10. Metody dla kolekcji

Opcja ta włącza możliwość korzystania z metod wykonywania operacji na tablicach (np. COUNT, DELETE(), EXISTS(), EXTEND(), FIRST, LAST, LIMIT, NEXT(), PRIOR(), TRIM()).

Przykład deklaracji tablic i ich wykorzystania:


```

CREATE PACKAGE test
AS
    TYPE dane_arr_typ IS TABLE OF VARCHAR2(20) INDEX BY BINARY_INTEGER;
    TYPE dane_test_arr_typ IS VARRAY(5) OF VARCHAR2(10);
END;
/
DECLARE
    nazwa      test.dane_arr_typ;
BEGIN
    nazwa(1) := 'dane1';
    nazwa(2) := 'dane2';
    nazwa(3) := 'dane3';
    DBMS_OUTPUT.PUT_LINE('Liczba elementów: ' || nazwa.COUNT);
    nazwa.DELETE(2);
    DBMS_OUTPUT.PUT_LINE('Liczba elementów: ' || nazwa.COUNT);
END;

```

4.11. Widoki zgodne ze słownikiem danych

Opcja ta umożliwia korzystanie z kompatybilnych widoków administracyjnych zgodnych ze słownikami danych Oracle. Widoki są tworzone automatycznie podczas generowania bazy danych. Realizowane jest tutaj wsparcie dla kompatybilnych z Oracle widoków słownika danych na poziomie bazy danych. Widoki te są tworzone w schemacie SYSIBMADM, dla danej bazy danych. Mamy tutaj kilka kategorii: ogólne, tabele i widoki, obiekty (procedury, wyzwalacze itp.), bezpieczeństwo.

Przykładowe widoki:

```

select * from dba_tables;
select * from user_tables;
select * from user_sys_privs;
select * from dictionary;

```

TABLE_SCHEMA	TABLE_NAME	COMMENTS
SYSIBMADM	USER_VIEWS	Description of the user's own views
SYSIBMADM	USER_VIEW_COLUMNS	Columns of user's views
SYSIBMADM	USER_TRIGGERS	Triggers owned by the user
SYSIBMADM	USER_TABLESPACES	Description of accessible tablespaces
SYSIBMADM	USER_TABLES	Description of the user's own relational tables
SYSIBMADM	USER_TAB_PRIVS_RECD	Grants on objects for which the user is the grantee
SYSIBMADM	USER_TAB_PRIVS_MADE	All grants on objects owner by the user
SYSIBMADM	USER_TAB_PRIVS	Grants on objects for which the user is the owner
SYSIBMADM	USER_TAB_PARTITIONS	Description of partitioned-level partitioning information for the user's own
SYSIBMADM	USER_TAB_COMMENTS	Comments on tables and views owned by the user
SYSIBMADM	USER_TAB_COLUMNS	Columns of user's tables
SYSIBMADM	USER_TAB_COL_STATI...	Columns statistics for columns of user's tables
SYSIBMADM	USER_SYS_PRIVS	System privileges granted to current user
SYSIBMADM	USER_SYNONYMS	The user's private synonyms
SYSIBMADM	USER_SOURCE	Source of stored objects accessible to the user
SYSIBMADM	USER_SEQUENCES	Description of the user's own SEQUENCES
SYSIBMADM	USER_ROLE_PRIVS	Roles granted to current user
SYSIBMADM	USER_PROCEDURES	Description of the user's own procedures

Rys. 3. Wynik zapytania na widoku DICTONARY
 Fig. 3. The result of a query on a view DICTONARY

Wynik działania ostatniego zapytania zwraca informacje o widokach operujących na metadanych, ich opis przedstawiono na rys. 3.

4.12. Kompilacja PL/SQL

Kod języka PL/SQL może być kompilowany i wykonywany przy użyciu interfejsów DB2. Pozwala to na wykorzystanie już istniejących rozwiązań napisanych w języku PL/SQL, które docelowo będą działać z serwerem danych DB2 (wsparcie PL/SQL musi być włączone na serwerze DB2). Obsługiwane interfejsy obejmują procesor wiersza komend (CLP), CLPPlus i IBM Data Studio. Włączenie tego parametru praktycznie dostarcza najwięcej funkcjonalności dla języka PL/SQL.

Włączenie tej opcji umożliwia:

- PL/SQL i skrypty mogą być kompilowane i wykonywane przy użyciu interfejsów DB2,
- tworzenie procedur PL/SQL i funkcji z linii komend CLP ze skryptu,
- wykorzystanie podstawowej składni, instrukcji sterujących występujących w PL/SQL,
- wykorzystanie bloków, procedur, funkcji, kolekcji, kursorów, pakietów, typów i zmiennych PL/SQL.

Występują tutaj także ograniczenia w stosunku do języka PL/SQL:

- funkcje i wyzwalacze nie mogą być tworzone w środowisku partycjonowanym (DPF – Database Partitioning Feature),
- nie są obsługiwane typy danych NCLOB i XMLTYPE,
- brak obsługi deklaracji typów TYPE w funkcjach, procedurach, wyzwalaczach i blokach anonimowych.

4.13. Insensitive cursor (dotyczy Sybase i SQL Server)

Opcja ta umożliwia definiowanie kursorów z opcją WITH RETURN, które są nieczułe na zmiany bez opcji FOR UPDATE (kursor tylko do odczytu). Zmienna rejestru DB2_COMPATIBILITY_VECTOR o wartości ORA nie włącza tej opcji i dotyczy systemu Sybase.

Przykład:

```
BEGIN
  DECLARE result INSENSITIVE CURSOR WITH RETURN TO CLIENT FOR
                                     SELECT * FROM TEST;
  OPEN result;
  DELETE FROM TEST;
  -- CLOSE result;
END@
```

4.14. Paramter INOUT (dotyczy Sybase i SQL Server)

Opcja ta umożliwia wykorzystanie specyfikacji DEFAULT dla deklaracji parametru INOUT. Deklaracja procedury pozwala na określenie wyrażenia DEFAULT dla parametru typu INOUT. Procedura INOUT, zdefiniowana z parametrem domyślnym, może być wywołana bez podania argumentów dla danego parametru. Jeśli argument dla parametru INOUT nie jest określony lub argument to słowo kluczowe DEFAULT, to dany parametr w procedurze nie zwraca żadnej wartości po wyjściu z procedury. Zmienna rejestru DB2_COMPATIBILITY_VECTOR o wartości ORA nie włącza tej opcji.

Przykład procedury z parametrem INOUT i jej wywołania:

```
CREATE OR REPLACE PROCEDURE test_INOUT
  (IN a1 INTEGER,
   IN a2 INTEGER,
   INOUT a3 INTEGER DEFAULT NULL) ...
/
CALL test_INOUT (1, 1, DEFAULT);
CALL test_INOUT (1, 1);
```

4.15. Poziomy dostęp do danych podczas wykonywania poleceń SQL

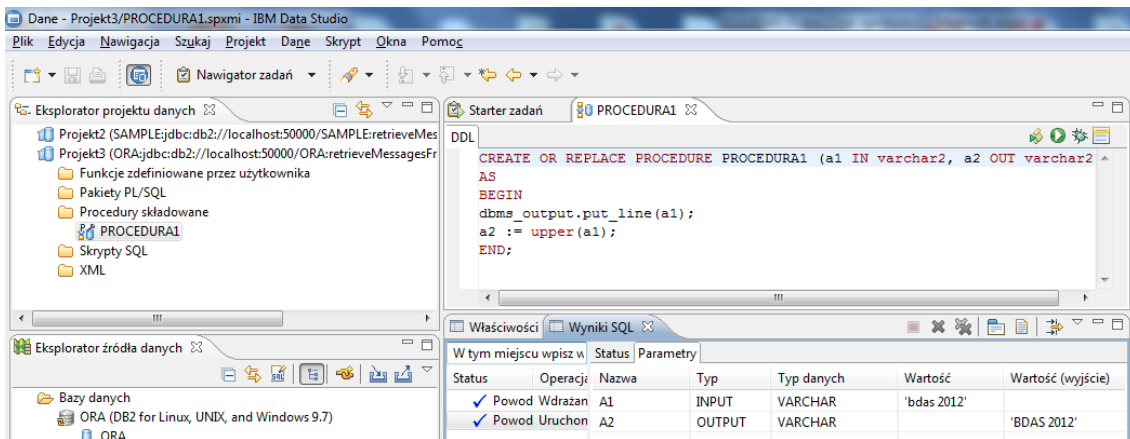
Stopień, w jakim procedury (procedury przechowywane lub funkcje zdefiniowane przez użytkownika, np. w języku JAVA) mogą wykonywać polecenia SQL, jest określony przez poziomy dostęp SQL.

Istnieją cztery poziomy dostęp SQL do danych:

- NO SQL (funkcja nie może wykonać żadnych poleceń SQL)
- CONTAINS SQL (funkcja zawiera polecenia SQL),
- READS SQL DATA (funkcja może czytać dane SQL),
- MODIFIES SQL DATA (funkcja może modyfikować dane SQL).

5. Narzędzia wspierające obsługę języka PL/SQL i migrację danych

Podstawowym narzędziem programisty baz danych DB2, wykorzystujących aplikacje Oracle, jest IBM Data Studio, który wspiera obsługę języka PL/SQL. Narzędzie to przedstawiono na rys. 4. Innym ważnym elementem jest migracja systemu Oracle do DB2. Chcemy docelowo wykonywać aplikacje Oracle z poziomu bazy danych DB2. Migrację taką możemy przeprowadzić za pomocą narzędzia IBM Data Movement Tool.



Rys. 4. IBM Data Studio

Fig. 4. IBM Data Studio

6. Podsumowanie

Artykuł prezentuje możliwość kompilacji i wykonywania aplikacji Oracle w systemie DB2. Przedstawia także sposób konfiguracji systemu DB2 do współpracy z językiem PL/SQL. Aby można było to zrealizować, wprowadzono wiele funkcji zgodności z Oracle i możliwość ich selektywnego wyboru za pomocą zmiennej rejestru. W jednym systemie pojawia się język SQL PL oraz język PL/SQL, wraz z całym zestawem pakietów i funkcjonalności stosowanych w systemie Oracle (dotyczy to również standardu języka SQL, który także musiał zostać dostosowany). Dzięki tym operacjom najnowszy darmowy DB2 9.7.5 Express-C pozwala na łatwe uruchamianie aplikacji Oracle w DB2, po wcześniejszej jej migracji, którą może wykonać za pomocą narzędzia IBM Data Movement Tool. Mamy docelowo możliwość wdrażania aplikacji na nowym serwerze baz danych DB2, zamiast na serwerze Oracle.

BIBLIOGRAFIA

1. Drzymała P., Welfle H.: Przetwarzanie zapytań rozproszonych w środowisku InfoSphere Federation Server stowarzyszonym z bazami danych Oracle. *Studia Informatica*, Vol. 32, No. 2B(97), Wydawnictwo Politechniki Śląskiej, Gliwice 2011, s. 359÷372.
2. Smarter Planet IBM, <http://www.ibm.com/smarterplanet/us/en/overview/ideas/>.
3. Chong R.F., Liu C., Qi C.F., Snow D.R.: *Understanding DB2: Learning Visually with Examples*.
4. IBM DB2 dla systemów Linux, UNIX i Windows – Centrum informacyjne, <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp>.

5. Chen W.-J., Chen K., Dantressangle P., Greenstein M., Lightstone S., Petrenko M., Reutlinger R., Rielau S., Schormann S., Schwenger M.N.: Oracle to DB2 Conversion Guide: Compatibility Made Easy. RedBooks, November 2009.

Wpłynęło do Redakcji 14 stycznia 2012 r.

Abstract

The article presents the mechanisms of support to run Oracle applications in DB2 9.7.5 Express-C. Presented the possibility compilation and execution applications in the DBMS DB2 9.7 Express-C using SQL, PL/SQL and Oracle objects. This version introduces support of Oracle 10g and 11g, which was previously available only in Enterprise Edition. In relation to older versions of commercial introduced new levels of compatibility with respect to Oracle. The article presents also how to configure DB2 to work with the language PL/SQL. Introduced a lot of functionality here. They can be selected using the variable DB2_COMPATIBILITY_VECTOR. In one system, use SQL PL and PL/SQL language, packages with built-in and functionality used in Oracle. Through these operations, the latest free DB2 Express-C 9.7.5 allows you to easily run Oracle applications to DB2 after its earlier migration, which can perform by using the IBM Data Movement Tool.

Adresy

Paweł Drzymała: Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych, ul. Stefanowskiego 18/22, 90-924 Łódź, Polska, pawel.drzymała@p.lodz.pl.

Henryk Welfle: Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych, ul. Stefanowskiego 18/22, 90-924 Łódź, Polska, henryk.welfle@p.lodz.pl.