

## SQL and Multidimensional Data

Conf.dr. Mihaela MUNTEAN  
Catedra de Informatică Economică, A.S.E. București

*Using SQL you can manipulate multidimensional data and extract that data into a relational table. There are many PL/SQL packages that you can use directly in SQL\*Plus or indirectly in Analytic Workspace Manager and OLAP Worksheet.*

*In this article I discussed about some methods that you can use for manipulating and extracting multidimensional data.*

**Keywords:** SQL, analytic workspace, multidimensional data.

**E**xistă o serie de pachete PL/SQL care pot fi utilizate pentru a manipula datele stocate în spațiile analitice și anume (figura 1):

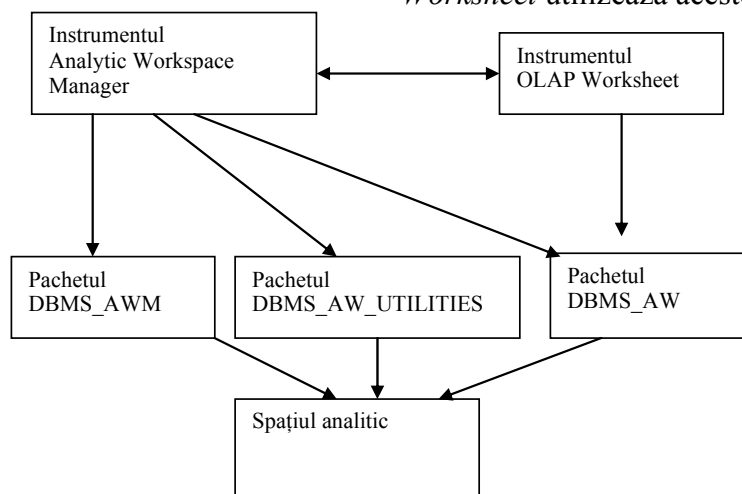
- pachetul *DBMS\_AW* utilizat pentru a executa comenzi de manipulare/definire OLAP asupra spațiilor analitice. De exemplu, se pot încărca date din tabele relaționale în spații analitice, se pot executa analize complexe pe datele stocate în spațiile analitice, iar rezultatele acestor analize pot fi stocate în tabele relaționale.

- pachetul *DBMS\_AWM* utilizat pentru a crea spații analitice în format standard.

- pachetul *DBMS\_AW\_UTILITIES* utilizat pentru a crea măsuri într-un spațiu analitic format standard. Pentru a utiliza acest pachet, utilizatorul trebuie să fie atașat rolurilor: *OLAP\_USER* și *DBA*.

- colecția de pachete *CWM2* utilizată pentru a defini catalogul de metadate OLAP.

Aceste pachete pot fi utilizate direct în *SQL\*Plus*. De asemenea, instrumentele *Analytic Workspace Manager* și *OLAP Worksheet* utilizează aceste pachete.



**Fig.1.** Pachete PL/SQL utilizate pentru a manipula datele stocate în spațiile analitice

Într-o comandă *SELECT-SQL* se poate utiliza funcția *OLAP\_TABLE()*, pentru a interoga datele stocate într-un spațiu analitic. Această funcție permite accesul la datele stocate în spațiul analitic și are următoarea sintaxă:

*OLAP\_TABLE(denumire\_spațiu\_analitic in varchar2, denumire\_tabela\_de\_obiecte in*

*varchar2, datamap in varchar2, limit\_map in varchar2)*

unde:

- *denumire\_spațiu\_analitic* este un parametru ce specifică denumirea spațiului analitic utilizat pentru interogare. Se specifică și perioada de conexiune la spațiul analitic (QUERY/SESSION);

- *denumire\_tabela\_de\_obiecte* este un parametru ce specifică denumirea unei tabeli de obiecte. Această tabelă este definită anterior și este utilizată pentru a structura datele multidimensionale în formatul bidimensional al tabelii relaționale;
- *datamap* este un parametru opțional, adesea omis și specifică o comandă OLAP ;
- *limit\_map* este un parametru utilizat pentru a mapa obiectele sursă din spațiul analitic (dimensiuni, măsuri) la coloanele corespunzătoare ale tabelii relaționale.

Pentru exemplificare, se va utiliza instrumentul *SQL\*Plus* și se face conexiunea la baza de date ca utilizator *utilizator\_aw*. Utilizatorul este atașat rolurilor *CONNECT*, *RESOURCE* și *OLAP\_USER* (este proprietarul spațiului analitic *GLOBAL*).

- Se va crea un tip de obiect *demo\_cub*. Un astfel de obiect este echivalent cu un tuplu al unei tabeli relaționale.

```
SQL> create type demo_cub as object
(canale varchar2(30), clienti varchar2(30),
 produse varchar2(30), timp
 varchar2(30), vanzari number);
```

- Se va crea o tabelă de obiecte *demo\_table* care va conține obiecte de tipul definit anterior:

```
SQL> create type demo_table as table of
demo_cub;
```

- Se face conexiunea la spațiul analitic *GLOBAL* cu drepturi numai de citire. Se utilizează procedura *execute* din pachetul *DBMS\_AW*:

```
SQL> execute dbms_aw.execute('aw attach
GLOBAL ro');
```

- Se execută comanda *LIMIT* prin care se stabilește setul de membrii pentru fiecare dimensiune și anume pentru: dimensiunea *Clienti* un membru (*client\_id=89*), dimensiunea *Canale* de distribuție un membru (*canal\_id=4*), dimensiunea *Produce* un membru (*produs\_id=48*) și dimensiunea *Timp* primul membru :

```
SQL> execute dbms_aw.execute('limit
global!clienti to "89");
```

```
SQL> execute dbms_aw.execute('limit
global!canale to "4");
```

```
SQL> execute dbms_aw.execute('limit
global!produse to "48");
```

```
SQL> execute dbms_aw.execute('limit
global!timp to first 1');
```

Măsura derivată *Vanzari* este deja creată în spațiul analitic. Se poate utiliza instrumentul *OLAP Worksheet* pentru a vizualiza valorile acestei măsuri (figura 2).

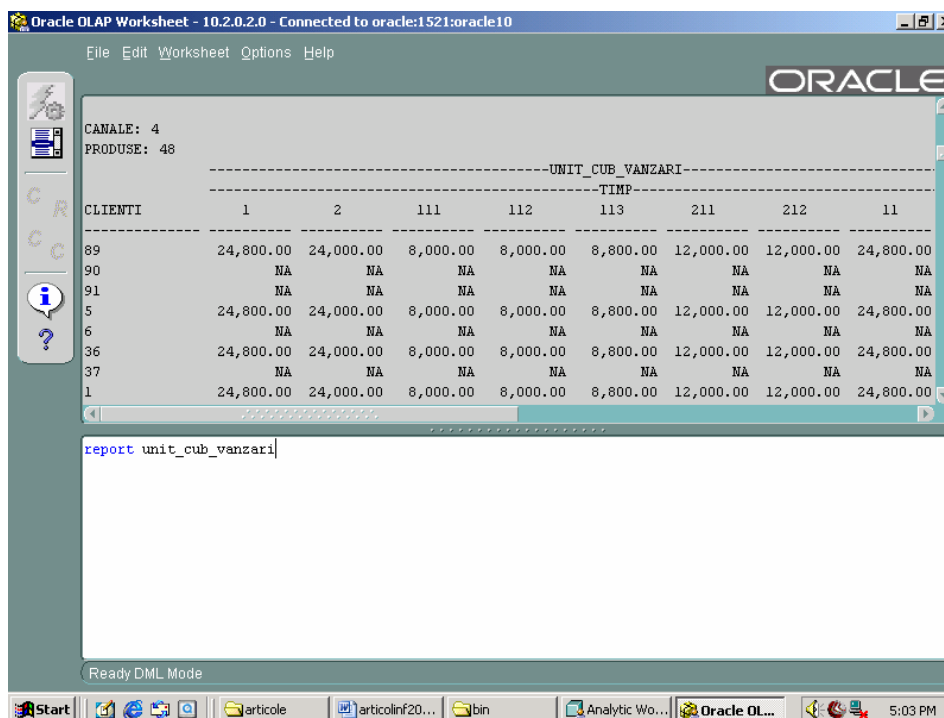


Fig.2. Utilizarea instrumentului OLAP Worksheet pentru vizualizarea valorilor măsurii derivate Vanzări

➤ Valorile măsurii *Vânzări* pot fi vizualizate și în *SQL\*Plus*, dacă se utilizează comanda *SELECT-SQL* ce apelează funcția *OLAP\_TABLE()*:

```
SQL>select canale, clienti, produse, timp,
vanzari
from table(OLAP_TABLE
```

```
('GLOBAL DURATION session',
'DEMO_TABLE', ",
'DIMENSION canale from global!canale
DIMENSION clienti from global!clienti
DIMENSION produse from global!produse
DIMENSION timp from global!timp
measure vanzari from
global!unit_cub_vanzari'));
```

CANALE	CLIENTI	PRODUSE	TIMP	VANZARI
4	89	48	1	24800

Parametrii funcției *OLAP\_TABLE()* specifică:

- denumirea spațiului analitic: *GLOBAL*;
- denumirea tabelii de obiecte: *DEMO\_TABLE*;
- măsura pe care dorim să o vizualizăm și dimensiunile ei.

➤ Valorile măsurii derivate *Vânzări* pot fi stocate într-o tabelă relațională. În acest scop, se va crea o tabelă *Vânzări* :

```
SQL>create table vanzari (timp
varchar2(30) not null, clienti varchar2(30)
not null,
```

```
produse varchar2(30) not null, canale
varchar2(30) not null,
vanzari number);
```

În spațiu analitic *GLOBAL* se va crea un program care va fi utilizat pentru a încărca în tabela *Vanzari* valorile măsurii *unit\_cube\_vanzari* (denumirea măsurii este precedată de denumirea cubului din care face parte). Dimensiunile măsurii sunt: *Timp*, *Canale*, *Clienti* și *Produse*. Se vor selecta din aceste dimensiuni, numai membrii frunză, utilizând comanda *LIMIT* (figura 3).

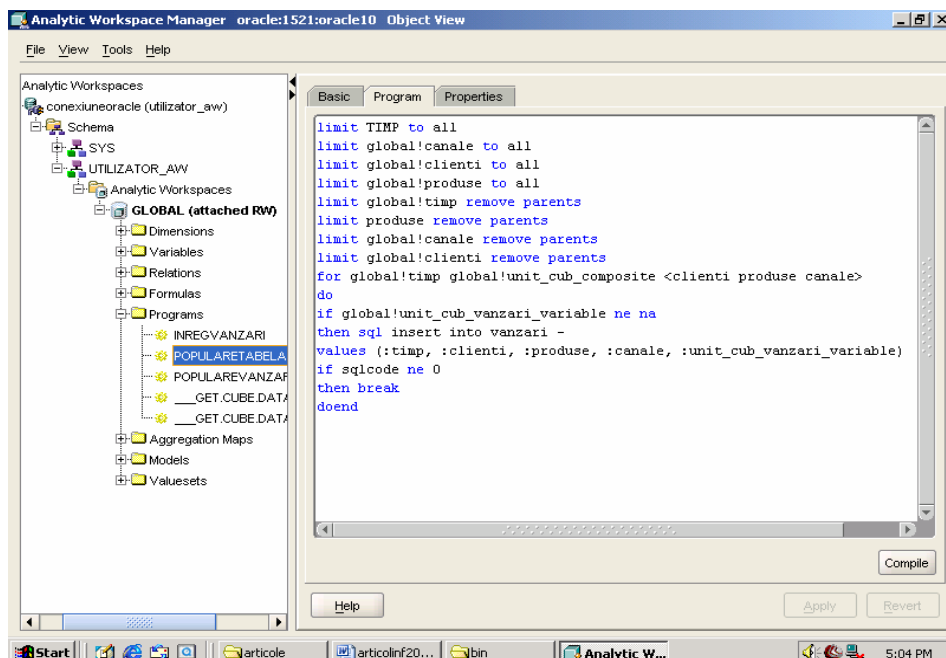


Fig.3. Programul utilizat pentru popularea unei tabeli cu date multidimensionale

Se execută programul *popularetabela* în *OLAP Worksheet*. Programul nu va încărca în tabela *Vânzări* și valorile null.

➤ Se utilizează apoi comanda *SELECT-SQL* în *SQL\*Plus* sau în *OLAP Worksheet* (figura 4) pentru a afișa tuplurile tabelii *Vânzări*:

SQL> select \* from vanzari;

TIMP	CLIENTI	PRODUSE	CANALE	VANZARI
111	89	48	4	8000
111	90	49	4	8000
111	89	50	4	10400
111	89	48	5	8800

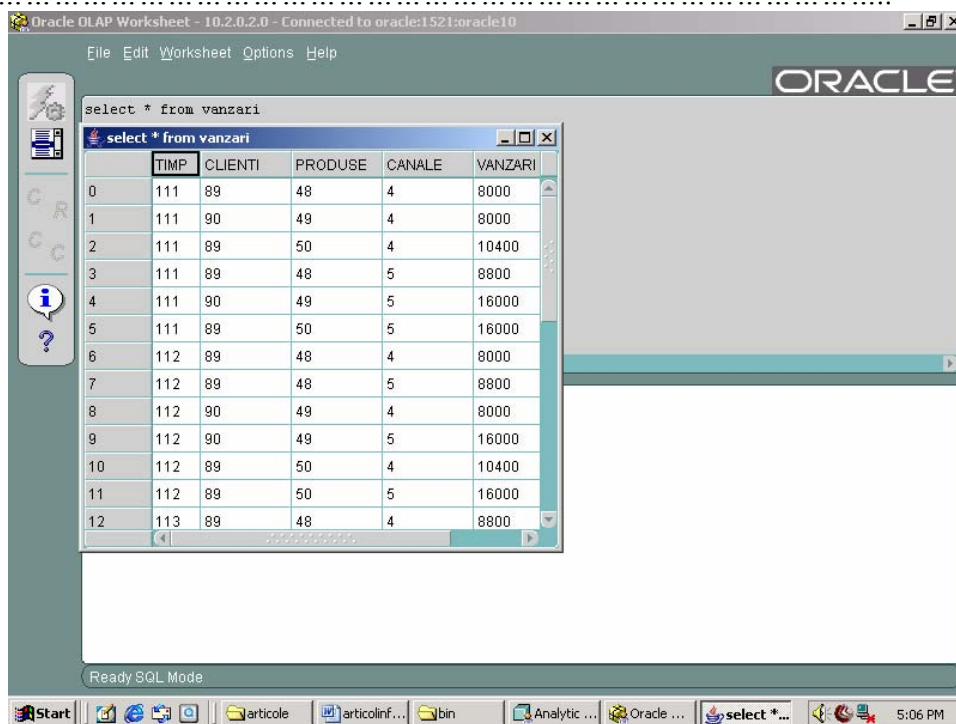


Fig.4. Utilizarea instrumentului OLAP Worksheet pentru a afișa tuplurile tabelii Vânzări

**Bibliografie**

[1] Mihaela Muntean, *Inițiere în tehnologia OLAP. Teorie și practică*, editura ASE, București, 2004

\*\*\*] Oracle Corporation, *Oracle OLAP Application Developer's Guide*, 2004  
 \*\*\*] Oracle Corporation, *Oracle OLAP Reference 10g Release 2*, 2005