UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

SPECIALIZAREA INFORMATICĂ

**DATA WAREHOUSE & BUSINESS INTELLIGENCE**    
**- PROIECT -**

Autor

Tiberius Coman

George Banica

Daniela Alexandra Constantin

Sebastian Alexandru Velciu

Profesor titular

Lect. Dr. GABRIELA MIHAI

   
GRUPA 405

ANUL I MASTER, SEMESTRUL I

7.

Vom creea obiectele dimension ruland urmatorul bloc de cod pentru a ajunge la rezultatul mentionat in partea de analiza:

-- create dimensions

DROP DIMENSION locatie;

CREATE DIMENSION locatie

LEVEL locatie\_id IS (dim\_locatie.id\_locatie)

LEVEL strada IS (dim\_locatie.strada) SKIP WHEN NULL

LEVEL oras IS (dim\_locatie.oras) SKIP WHEN NULL

LEVEL tara IS (dim\_locatie.tara) SKIP WHEN NULL

HIERARCHY ierarhie\_locatie (locatie\_id CHILD OF strada CHILD OF oras CHILD OF tara)

ATTRIBUTE locatie\_info LEVEL locatie\_id DETERMINES

(dim\_locatie.strada, dim\_locatie.oras, dim\_locatie.tara, dim\_locatie.site);

DROP DIMENSION timp;

CREATE DIMENSION timp

LEVEL data\_id IS (dim\_calendar.id\_data)

LEVEL data IS (dim\_calendar.data)

LEVEL anul IS (dim\_calendar.anul)

HIERARCHY ierarhie\_timp (data\_id CHILD OF data CHILD OF anul)

ATTRIBUTE timp\_id\_info LEVEL data\_id DETERMINES

(dim\_calendar.id\_data, dim\_calendar.data, dim\_calendar.ziua, dim\_calendar.luna, dim\_calendar.anul)

ATTRIBUTE timp\_data\_info LEVEL data DETERMINES

(dim\_calendar.id\_data, dim\_calendar.data, dim\_calendar.ziua, dim\_calendar.luna, dim\_calendar.anul);

Pentru a putea vizualiza dimensiunile creeate in baza de date, vom rula:

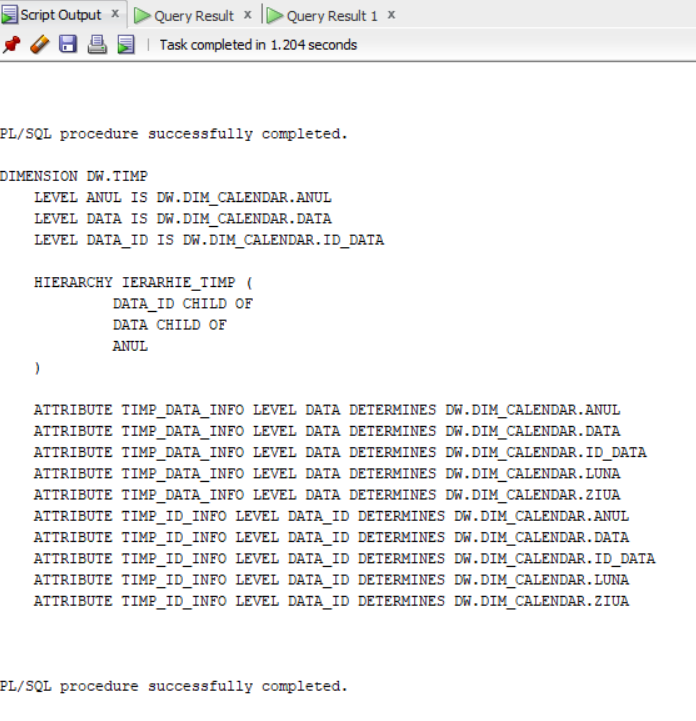
-- display dimensions

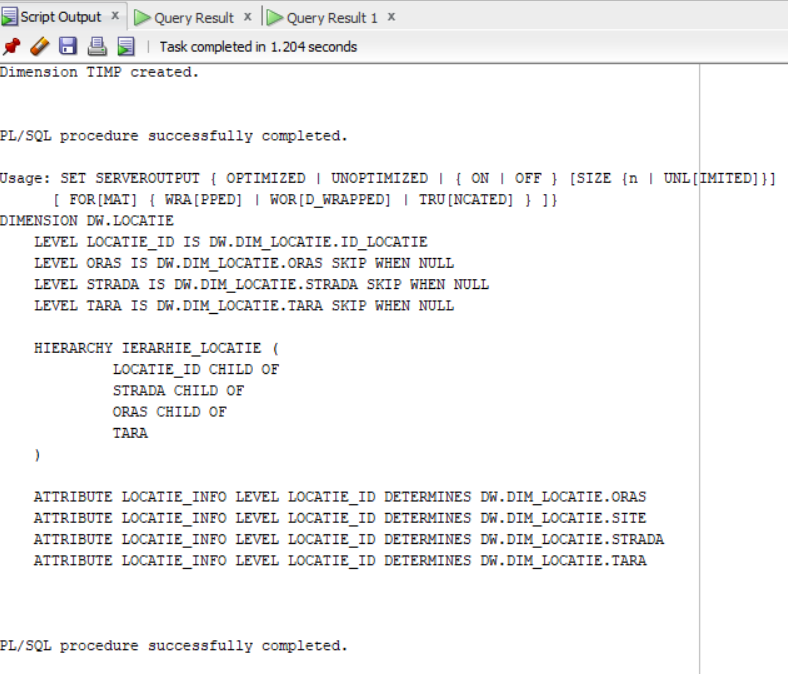
SET SERVEROUTPUT ON FORMAT WRAPPED; --to improve the display of info

EXECUTE DBMS\_DIMENSION.DESCRIBE\_DIMENSION('locatie');

EXECUTE DBMS\_DIMENSION.DESCRIBE\_DIMENSION('timp');

La rulare o sa fie afisate mesajele urmatoare, semn ce inseamna ca obiectele au fost creeate cu succes:





Pentru a valida dimensiunile vom rula urmatoare bucata de cod:

-- validate dimensions

-- used to create dimensions exceptions table

@utldim.sql

EXECUTE DBMS\_DIMENSION.VALIDATE\_DIMENSION ('locatie', FALSE, TRUE, 'validare locatie');

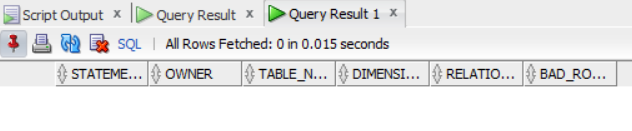
EXECUTE DBMS\_DIMENSION.VALIDATE\_DIMENSION ('timp', FALSE, TRUE, 'validare timp');

Pentru a verifica exceptiile, vom folosi urmatoarea instructiune:

-- verificarea exceptiilor

SELECT \* FROM dimension\_exceptions;

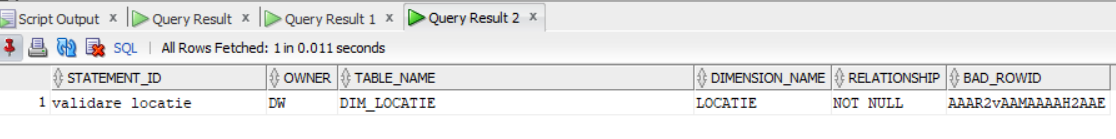
Rezultatul ar trebui sa fie un tabel gol, in caz contrar, inseamna ca dimensiunile nu au fost definite corect.



Deoarece a fost adaugata instructiunea „SKIP WHEN NULL” la creearea dimensiunii locatie, iar tabelul time contine restrictii, nu vom avea exceptii.

Vom analiza cazurile in care aceste 2 dimensiuni ar fi fost create gresit, ce exceptii am fi primit si cum le puteam interpreta.

In primul rand, daca nu am fi introdus instructiunea „SKIP WHEN NULL” si am fi verificat tabelul de exceptii dupa validarea dimensiunilor, acesta ar fi aratat astfel:



Observam ca aceste coloane nu ne ajuta foarte mult in a gasi eroarea, secretul consta insa in a „match-ui” coloana „bad\_rowid” cu cea din tabelul nostru („dim\_locatie”).

Vom rula urmatorul set de instructiuni pentru a primi un mesaj al erorii mai descriptiv:

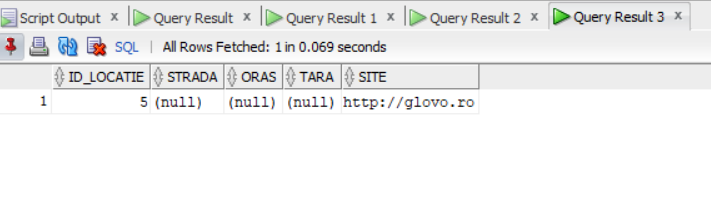
-- match-uirea exceptiilor in tabela pentru identificarea randurilor cu probleme

SELECT \* FROM dim\_locatie

WHERE rowid IN (SELECT bad\_rowid

FROM dimension\_exceptions

WHERE statement\_id = 'validare locatie');



Astfel, observam ca a fost aruncata exceptia la rularea dimensiunii locatie, deoarece cele 3 coloane („strada”, „oras”, „tara”) pot fi nule in anumite situatii, mai exact cand cumparatura este executata online, prin intermediul unui site, in locul unei locatii fizice de la un comerciant.

Un alt element cheie de observat este legat de creearea dimensiunii timp si anume faptul ca aceasta nu contine nivelurile „zi” si „luna”. Motivul este foarte simplu, pentru a creea o dimensiune corect, regula de baza in ierarhie este ca relatiile sa fie de tipul 1:n. Regula de baza spune ca: „Trebuie să existe o relație 1:n între un părinte și copii. Un părinte poate avea unul sau mai mulți copii, dar un copil poate avea un singur părinte.”

Prin urmare daca am fi introdus nivelele „ziua” si „luna”, am fi primit ca exceptie toate randurile din tabelul „dim\_calendar”. Urmatoarea secventa de cod si captura de ecran vor demonstra aceasta explicatie.

DROP DIMENSION timp;

CREATE DIMENSION timp

LEVEL data\_id IS (dim\_calendar.id\_data)

LEVEL data IS (dim\_calendar.data)

LEVEL ziua IS (dim\_calendar.ziua)

LEVEL luna IS (dim\_calendar.luna)

LEVEL anul IS (dim\_calendar.anul)

HIERARCHY ierarhie\_timp (data\_id CHILD OF data CHILD OF ziua CHILD OF luna CHILD OF anul)

ATTRIBUTE timp\_id\_info LEVEL data\_id DETERMINES

(dim\_calendar.id\_data, dim\_calendar.data, dim\_calendar.ziua, dim\_calendar.luna, dim\_calendar.anul)

ATTRIBUTE timp\_data\_info LEVEL data DETERMINES

(dim\_calendar.id\_data, dim\_calendar.data, dim\_calendar.ziua, dim\_calendar.luna, dim\_calendar.anul);

Selectam erorile cu:

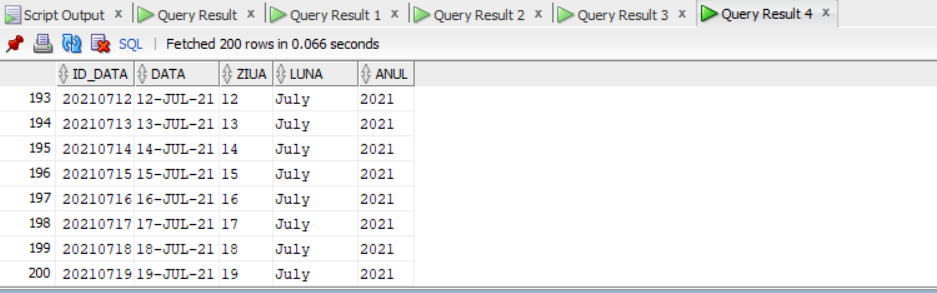
SELECT \* FROM dim\_calendar

WHERE rowid IN (SELECT bad\_rowid

FROM dimension\_exceptions

WHERE statement\_id = 'validare timp');

Rezultatul va fi format din toate randurile din tabelul „dim\_calendar”:



La final, tabelul de exceptii poate fi curatat folosind instructiunea:

-- stergerea exceptiilor vechi (doar daca este necesar)

DELETE (SELECT \* FROM dimension\_exceptions);