

# TABLIČNE FUNKCIJE NA BAZI ZA BOLJE PERFORMANSE ETL-A

Nataša Dvoršak, Projektant IS  
ULJANIK IRI d.o.o.

Rovinj, 19.10.2012.g.

- Paralelne mogućnosti izvršenja SQL-a
- Tablične funkcije
- MapReduce implementacija

## ULJANIK Informatika, razvoj, investicije d.o.o.



- članica Grupe Uljanik
- zaposlenih 40
- djelatnosti: informatička i investicijska
  - **Informacijski sustavi: 16**
    - IS – razvoj, implementacija, održavanje, sistem integracija; poslovna partnerstva
  - **Informacijska tehnologija: 13**
    - IT – izgradnja, održavanje, upravljanje, administracija, sigurnosna zaštita
  - **Investicije: 7**
  - **Uprava: 4**
- 49 godina informatičke djelatnosti
- znanje, iskustvo, stručnost

**49**  
Godina  
**INFORMATIKE**  
1963.-2012.

[www.uljanik.hr](http://www.uljanik.hr)

## Proizvodi

- BURIN 
- BURINplus
- PRAMAR 
- MARS 
- Programska sučelja
  - MARS
  - PRIMAVERA 
  - UCM 

## Područja primjene

- Proizvodnja
  - Brodogradnja
  - Ostale industrije
- Uslužne djelatnosti

## Reference

- ULJANIK Grupa
- Grupa TESU
- 3.MAJ Brodogradilište
- Holcim
- Računovodstveni servis

# Paralelno izvršenje



## ■ Parallel Execution

- Parallel query
- Parallel DML
- Parallel DDL

# Princip paralelnog načina rada

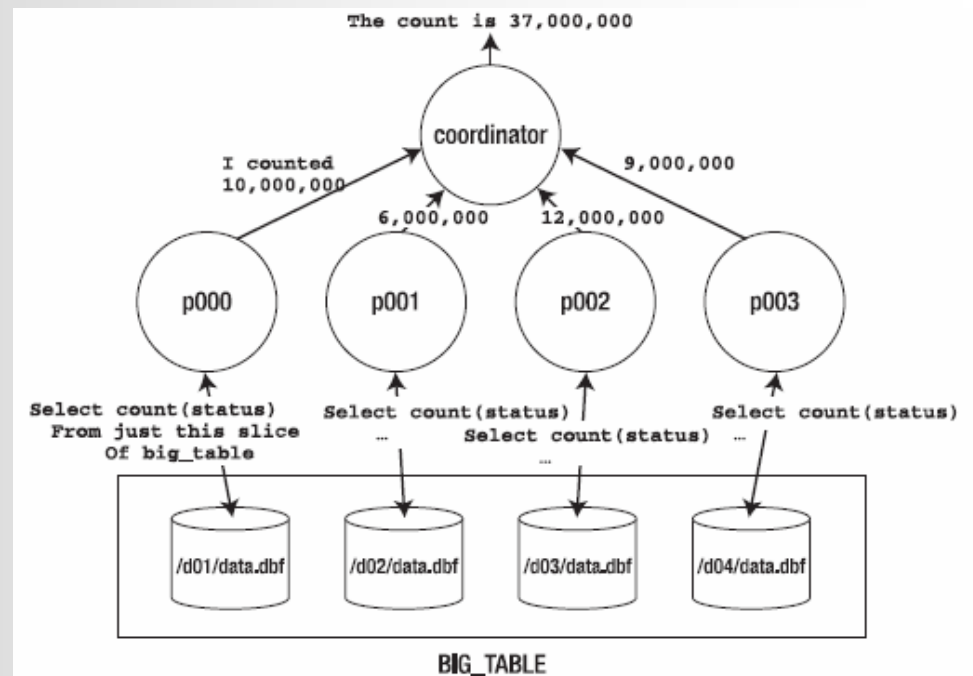


## ■ Primjer SQL upita

```
select /*+ parallel (t,10)*/ count(status) from big_table t;
```

## ■ Podaci u katalogu

```
select sid, username, program  
from v$session  
where sid  
in (select sid  
from v$px_session  
where qcsid = 250);
```





## ■ Degree of parallelism (DOP)

### ■ Naredba

Uporabom hinta PARALLEL ili PARALLEL\_INDEX, npr.

```
select /*+parallel(emp,4)*/ count(*) from emp;  
SELECT /*+PARALLEL_INDEX(mytab, myindex, 3)*/ stuff FROM mytab;
```

### ■ Instanca

Izmjenom parametara sesije, npr.

```
alter session force parallel query;
```

### ■ Objekt

Izmjenom u definiciji tablice ili indeksa npr.

```
alter table emp parallel (degree 4);  
CREATE INDEX myidx ON customer(sex, hair_color, customer_id)  
  PARALLEL 35;  
ALTER INDEX myidx parallel 35;
```

# Paralelno izvršenje SQL upita



## ■ SQL upit treba zadovoljiti uvjete

- Bar jednoj tablici se pristupa s  
“full table scan” ili “index range scan”

### ■ Za “full table scan” potrebno je:

#### → Koristiti hint

```
select /*+ parallel(emp,4) */ count(*) from emp;
```

#### → ili, izmijeniti definiciju tablice

```
alter table emp parallel (degree 4);
```

### ■ Za “index range scan” potrebno je:

#### → Da se upit grana na više particija

#### → Koristiti hint

```
select /*+parallel_index (emp,emp_idx,4)*/ count(*) from emp;
```

#### → ili, izmijeniti definiciju indeksa

```
alter index emp_idx parallel 4;
```



## ■ Eksplicitno zatražiti u sesiji

```
alter session enable parallel dml;
```

## ■ Pravila za UPDATE i DELETE

- Tablice moraju biti particionirane
- U operaciju mora biti uključeno više particija

## ■ Pravila za INSERT

- INSERT ... SELECT ... FROM

## ■ Nuparticionirane tablice ili indeksi

CREATE TABLE...AS SELECT

CREATE INDEX

ALTER INDEX...REBUILD

## ■ Partitionirane tablice ili indeksi

CREATE TABLE...AS SELECT

ALTER TABLE...MOVE PARTITION

ALTER TABLE...SPLIT PARTITION

CREATE INDEX

ALTER INDEX...REBUILD PARTITION

ALTER INDEX...SPLIT PARTITION

## ■ Ograničenje

LOB ili objektne kolone

## ■ Parallel Execution Servers

- Automatsko podešavanje
- `parallel_min_servers`, `parallel_max_servers`

## ■ Job Queue Processes

- `dbms_job`, `dbms_scheduler`
- Asinkrono izvođenje
- Red čekanja
- `job_queue_processes`

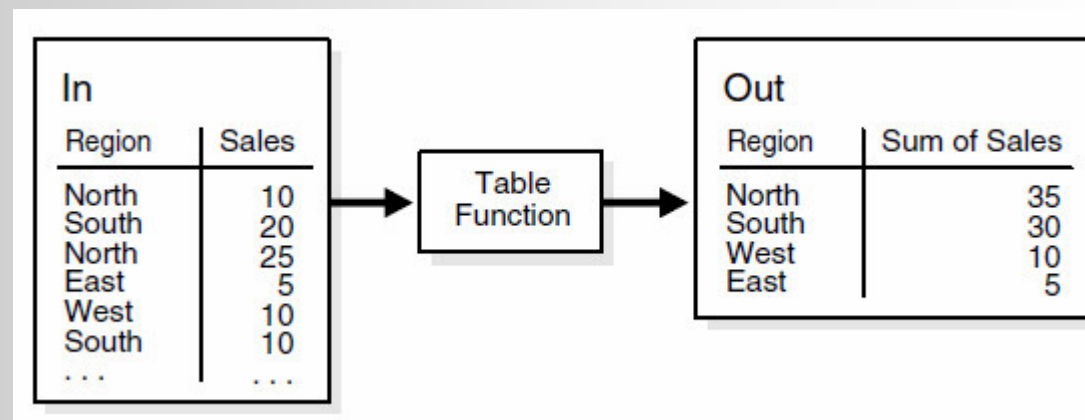
## ■ User Session Processes

- “Ručno”

## ■ Osnovne karakteristike

- Rezultat je skup redova (SQL tip na bazi)
- Koriste se kao tablice u FROM klauzuli
- Rezultat puni čitavu kolekciju redaka
- Ulaz može biti REF CURSOR

```
INSERT INTO Out SELECT * FROM TABLE(Table_Function(SELECT * FROM In));
```



# Tablične funkcije - primjer



```
-- Create the types to support the table function.
CREATE TYPE t_tf_row AS OBJECT (
  id NUMBER,
  description VARCHAR2(50) );
/
CREATE TYPE t_tf_tab IS TABLE OF t_tf_row;
/
-- Build the table function itself.
CREATE OR REPLACE FUNCTION get_tab_tf (p_rows IN NUMBER) RETURN t_tf_tab AS
  l_tab t_tf_tab := t_tf_tab();
BEGIN
  FOR i IN 1 .. p_rows LOOP
    l_tab.extend;
    l_tab(l_tab.last) := t_tf_row(i, 'Description for ' || i);
  END LOOP;
  RETURN l_tab;
END;
/

SELECT *
FROM TABLE(get_tab_tf(5)) ORDER BY id DESC;
ID          DESCRIPTION
-----
5           Description for 5
4           Description for 4
3           Description for 3
2           Description for 2
1           Description for 1
```

## ■ Osnovne karakteristike

- Vraćaju rezultat “red po red”

- Izlaz može biti PL/SQL tip

## ■ Sintaksa

- Klauzula PIPELINED

- PIPE ROW za vraćanje rezultata “red po red”

- Završava praznom RETURN naredbom



# Pipelined tablične funkcije - primjer



```
CREATE OR REPLACE FUNCTION get_tab_tf (p_rows IN NUMBER) RETURN
  t_tf_tab
PIPELINED AS
BEGIN
  FOR i IN 1 .. p_rows LOOP
    PIPE ROW(t_tf_row(i, 'Description for ' || i));
  END LOOP;
  RETURN;
END;
```

```
SELECT *
FROM TABLE(get_tab_tf(5)) ORDER BY id DESC;
```

ID	DESCRIPTION
5	Description for 5
4	Description for 4
3	Description for 3
2	Description for 2
1	Description for 1

## ■ Funkcija treba zadovoljiti uvjete

- PARALLEL\_ENABLE
- REF CURSOR ulazni parametar
- PARTITION BY
- REF CURSOR parametar treba imati DOP

```
CREATE FUNCTION function-name (parameter-name ref-cursor-type)
RETURN rec_tab_type PIPELINED
PARALLEL_ENABLE (PARTITION parameter-name BY [{HASH|RANGE} (column-list) | ANY])
[ORDER | CLUSTER] parameter-name BY (column-list) IS
BEGIN
...
END;
```

# Podjela podataka po instancama

## ■ PARTITION BY ANY

- Slučajnim odabirom redovi se ravnomjerno raspoređuju
- Kursori slabog tipa mogu koristiti samo ovu opciju



## ■ PARTITION BY HASH

- Formiranje podskupova prema zadanim kolonama
- Čitav podskup dodjeljuje se jednoj instanci
- Jedna instanca može zaprimiti i nekoliko podskupova
- Uravnotežena podjela podataka



## ■ PARTITION BY RANGE

- Formiranje podskupova prema zadanim kolonama
- Prema rastućem redosljedu
- Neravnomjerna distribucija podataka

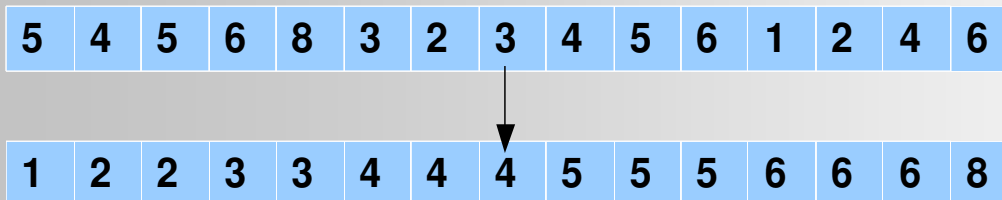


## ■ Uređivanje podataka instance

### ■ CLUSTER



### ■ ORDER

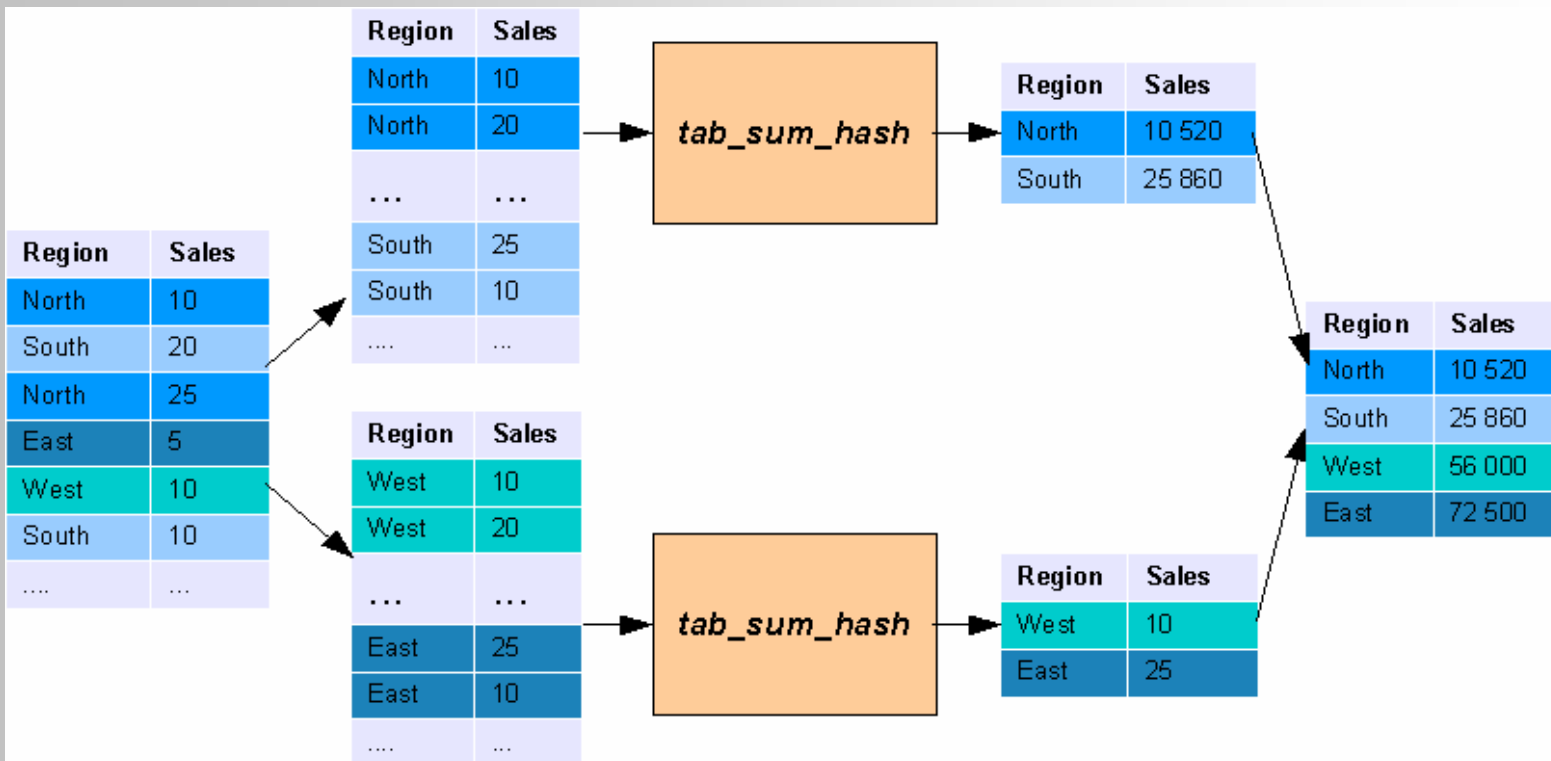


### ■ Uređivanje nema smisla za PARTITION BY ANY

# Paralelno izvršenje tablične funkcije

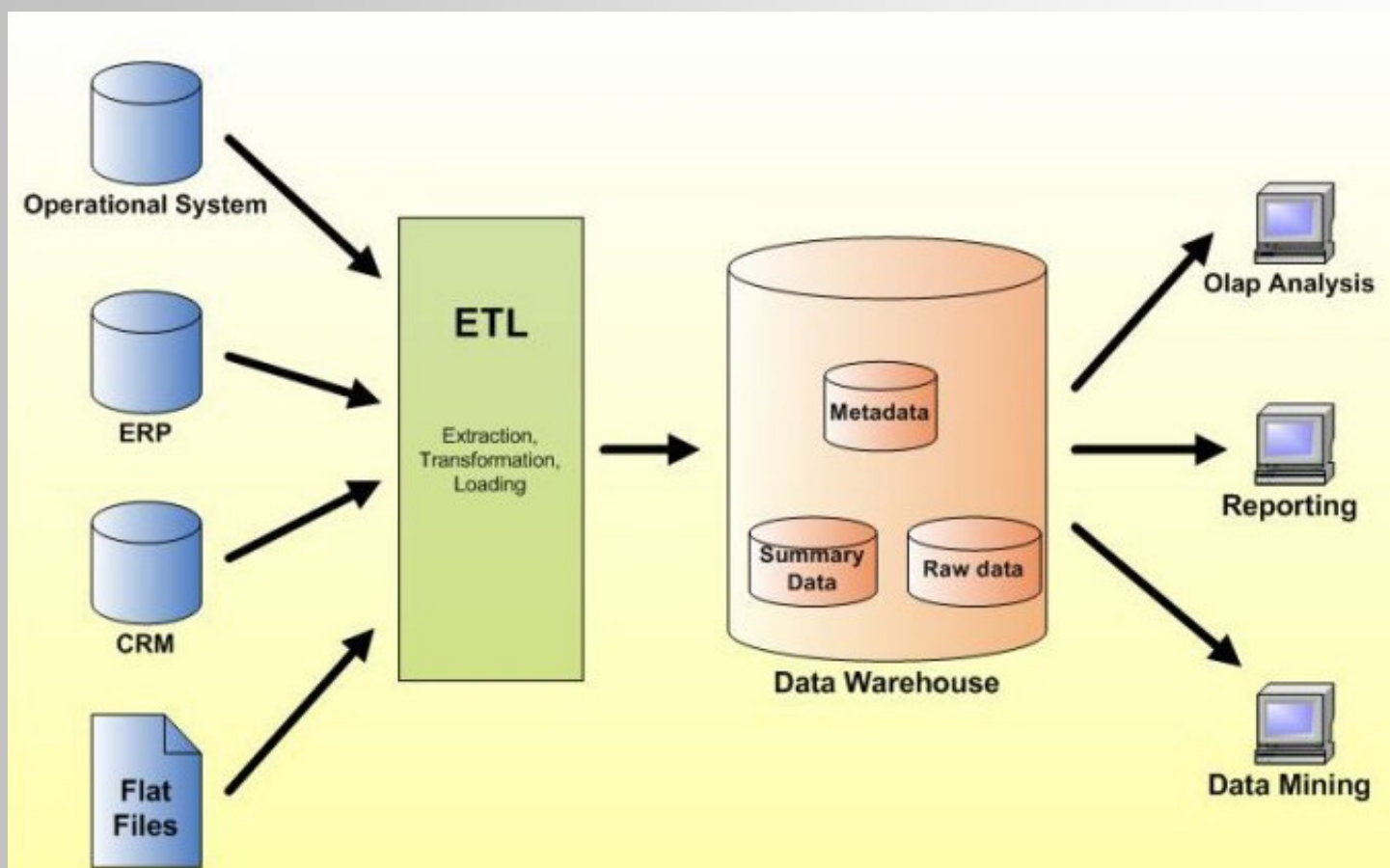


```
FUNCTION tab_sum_hash (p_cursor IN t_ref_cursor)
RETURN t_tab PIPELINED
PARALLEL_ENABLE(PARTITION p_cursor BY HASH (region))
CLUSTER p_cursor BY (region);
```



```
SELECT * FROM TABLE(tab_sum_hash (SELECT /*+parallel (s,10)*/ s.*
FROM sales s));
```

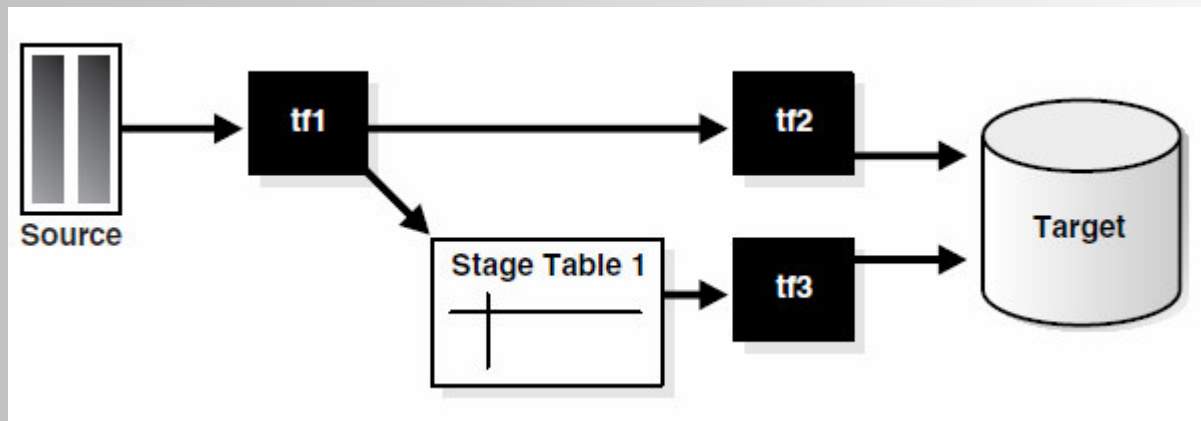
# Primjena u ETL procesu





## ■ Tablične funkcije u ETL procesu

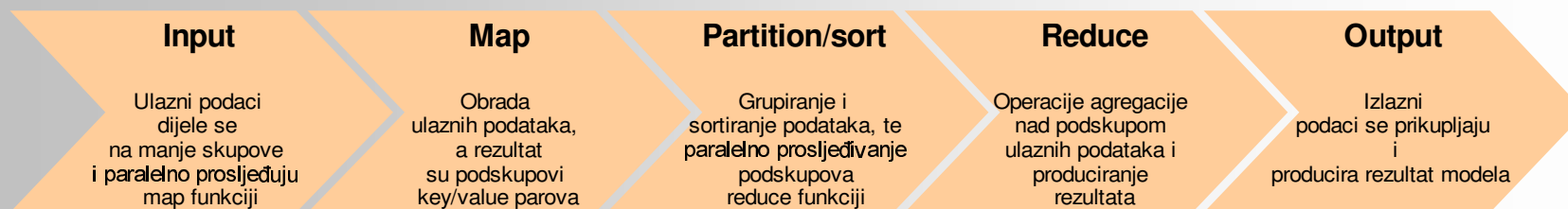
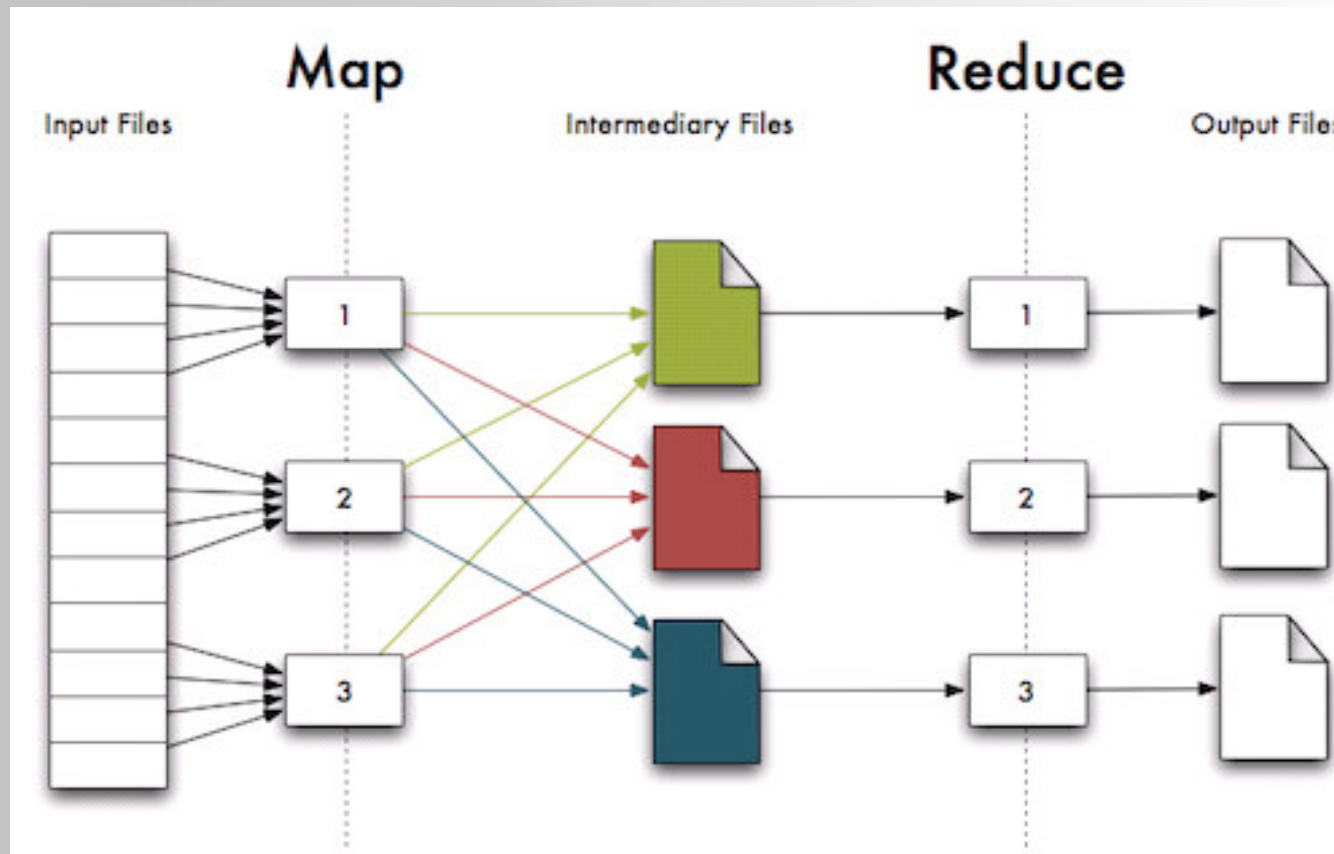
- Vraćaju redove kao rezultate
- Rezultati SQL upita ili tablične funkcije mogu biti ulazni CURSOR parametri
- Skupovi izlaznih podataka vraćaju se postepeno
- Izvršenje se može paralelizirati



```
insert into target
SELECT * FROM TABLE(tf2 (SELECT * FROM TABLE(tf1 (SELECT * FROM source))));

INSERT INTO target SELECT * FROM TABLE(tf3(SELECT * FROM stage_table1));
```

# MapReduce model



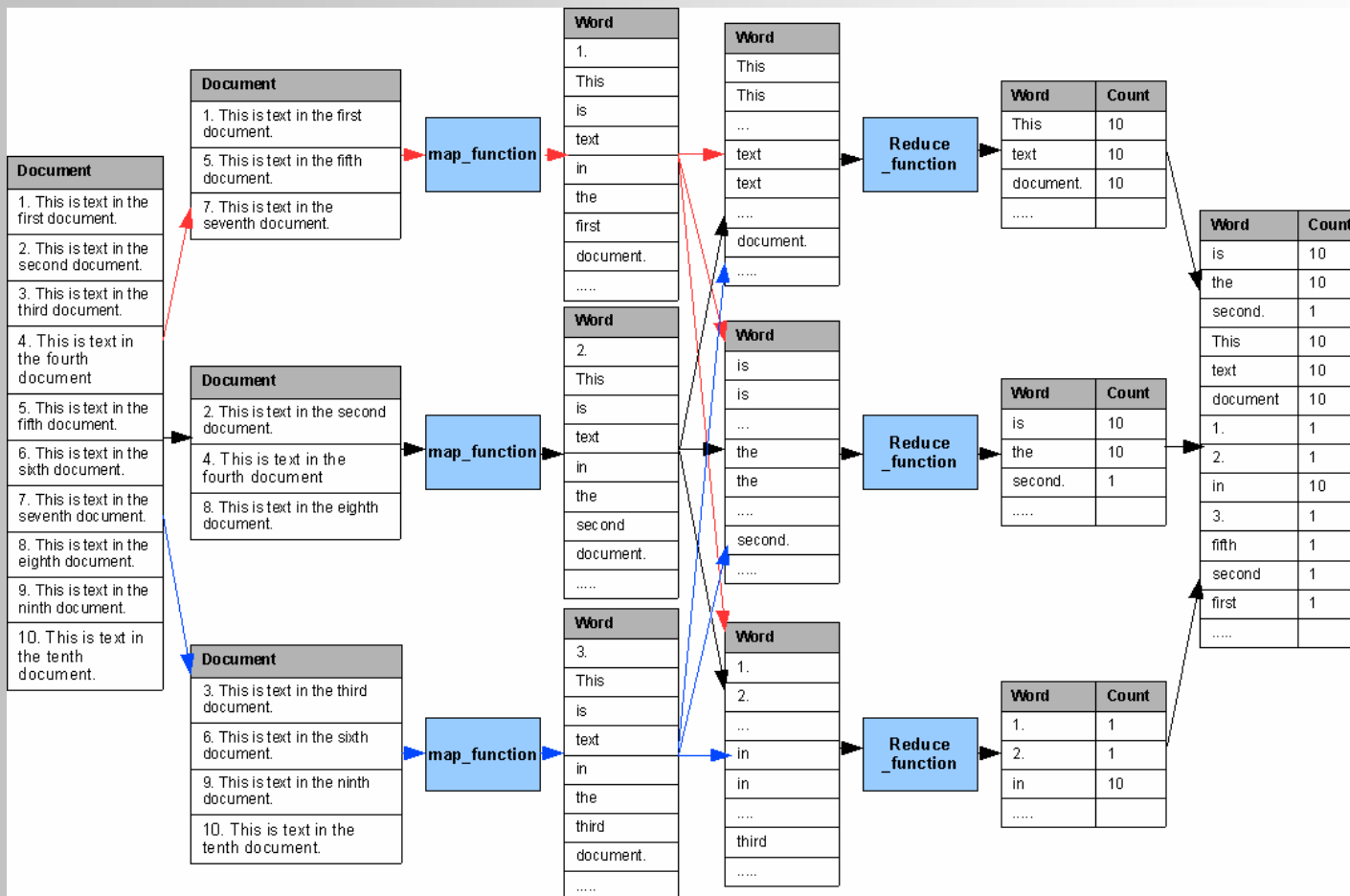
# MapReduce implementacija

```
SELECT *
```

```
FROM TABLE(reduce_function (CURSOR(SELECT *
```

```
FROM TABLE(map_function (CURSOR(SELECT /*+parallel (i,10)*/ i.*
```

```
FROM input_file i)))));
```



- Paralelni rad troši više resursa
- Kompleksne transformacije
- Extensible optimizer
  - Procjena kardinalnosti za CBO
- Oracle primjena tabličnih funkcija
  - `dbms_data_mining`

**Expert PL/SQL Practices: for Oracle Developers and DBAs**, by John Beresnewicz, Adrian Billington, Martin Büchi, Melanie Caffrey, Ron Crisco, Lewis Cunningham, Dominic Delmolino, Sue Harper, Torben Holm, Connor McDonald, Arup Nanda, Stephan Petit, Michael Rosenblum, Robyn Sands, Riyaj Shamsudeen, Apress (2011):

**Oracle PL/SQL Programming, FIFTH EDITION**, by Steven Feuerstein, O'Reilly Media (2009)

**Expert Oracle Database Architecture Oracle Database 9i, 10g, and 11g Programming Techniques and Solutions Second Edition**, by Thomas Kyte, Apress

<http://www.oracle-base.com/articles/misc/pipelined-table-functions.php>

# Na kraju ...





Nataša Dvoršak, Projektant IS

**ULJANIK IRI d.o.o.**

[natasa.dvorsak@uljanik.hr](mailto:natasa.dvorsak@uljanik.hr)