



# UNIVERSIDADE ESTADUAL DE SANTA CRUZ (UESC)

Criada pela Lei 6.344, de 05.12.1991,  
e reorganizada pela Lei 6.898, de 18.08.1995 e  
pela Lei 7.176, de 10.09.1997

## CET091 – Banco de Dados II

Prof. Dr. Marcelo Ossamu Honda

Departamento de Ciências Exatas e Tecnológicas (DCET)  
mohonda(at)uesc(.)br

# Índices Parciais

- Abranger um subconjunto de dados pertencente a tabela;
  - Declaração de cláusula WHERE;

```
idx=# CREATE INDEX IDX_PESCADOR_NOME ON T_PESCADOR(TP_NOME)
idx=# WHERE TP_NOME LIKE 'Ad%';
CREATE INDEX
idx=#
```

# Índices Parciais

```
idx=# explain select * from t_pescador where tp_nome like 'W%';
               QUERY PLAN
-----
Seq Scan on t_pescador  (cost=0.00..51.51 rows=37 width=12)
  Filter: ((tp_nome)::text ~~ 'W% '::text)
(2 registros)

idx=# █
```

```
idx=# explain select * from t_pescador where tp_nome like 'Ad%';
               QUERY PLAN
-----
Bitmap Heap Scan on t_pescador  (cost=8.49..25.33 rows=67 width=12)
  Recheck Cond: ((tp_nome)::text ~~ 'Ad% '::text)
  -> Bitmap Index Scan on idx_pescador_nome  (cost=0.00..8.48 rows=67 width=0)
(3 registros)

idx=# █
```

Cluster

# Tabelas Heap

- Padrão de armazenamento no PostgreSQL;
  - Dados são armazenados sem ordem;
- `select * from t_pescador;`

tp_id	tp_nome	tp_sexo
11492	Adela	F
11459	Alber	M
11626	Fabri	M
11488	Anaed	M
483	Soliv	M
11838	Dilce	M
11127	Crist	M
1243	Maria	F
11504	Aless	M

# Tabelas Organizadas por Índices

- Index-Organized Tabs (IOT);
  - Tabelas são fisicamente reorganizadas por índices;
    - Não são operações “*one-time*”;
      - As alterações não são realizadas em cada operação;
    - Os índices devem previamente serem definidos;
      - As informações dos *clusters* são armazenadas no SGBD;

# Criando o Cluster

- `select * from t_pescador;`

tp_id	tp_nome	tp_sexo
11492	Adela	F
11459	Alber	M
11626	Fabri	M
11488	Anaed	M
483	Soliv	M
11838	Dilce	M
11127	Crist	M
1243	Mariz	F
11504		

```
idx=# create index idx_nome on t_pescador(tp_nome);  
CREATE INDEX  
idx=#  
idx=#  
idx=# cluster t_pescador using idx_nome;  
CLUSTER  
idx=#
```

# Verificando o Cluster

tp_id	tp_nome	tp_sexo
11573		M
1188	Abel	M
1604	Adail	F
1094	Adail	M
11291	Adail	M
11674	Adalv	M
310	Adaut	M
1005	Adeci	F
1120	Adeil	M
11992	Adeil	F
11492	Adela	F
959	Adelc	M
979	Adeli	F



# Mantendo o Cluster

```
idx=# INSERT INTO T_PESCADOR VALUES (20000,'AA','M');  
INSERT 0 1  
idx=#  
idx=# INSERT INTO T_PESCADOR VALUES (20001,'Ab','M');  
INSERT 0 1  
idx=#
```

12018	Zica	M
12120	Zilda	F
416	Zilda	F
11172	Zilda	F
11456	Zildo	M
1686	Zito	M
224	Zorro	M
1850	Zulei	F
20000	AA	M
20001	Ab	M

(2841 registros)

# Atualizando o Cluster

```
idx=# CLUSTER T_PESCADOR;  
CLUSTER  
idx=#  
idx=# █
```

tp_id	tp_nome	tp_sexo
11573		M
20000	AA	M
20001	Ab	M
1188	Abel	M
1604	Adail	F
1094	Adail	M
11291	Adail	M
11674	Adalv	M
310	Adaut	M
1005	Adeci	F

# Reindexação

# Reindexação

- Busca efetivar o desempenho do index;

```

                                Lista de relações
Esquema | Nome | Tipo | Dono
-----+-----+-----+-----
public | t_pescador | tabela | mohonda
public | t_pescador_has_t_porto | tabela | mohonda
public | t_porto | tabela | mohonda
(3 registros)

idx=# \d t_pescador
          Tabela "public.t_pescador"
Coluna | Tipo | Modificadores
-----+-----+-----
tp_id | integer | não nulo
tp_nome | character varying(60) | não nulo
tp_sexo | character varying(1) |
Índices:
    "idx_nome" btree (tp_nome) CLUSTER
    "idx_pescador_nome" btree (tp_nome) WHERE tp_nome::text ~~ 'Ad% '::text

idx=# reindex table t_pescador;
REINDEX
idx=# reindex index idx_nome;
REINDEX
idx=# █
```

Vacuum

# Vacuum


- Recupera os espaços alocados por tuplas mortas;
  - Analogamente a um garbage collector;
  - Full: todo o banco;
  - Verbose: e listado os detalhes;
  - Analyse: atualiza as estatísticas do banco de dados;
  - <relação>: especifica a tabela;
  - <coluna>



# Vacuum

```
idx=# vacuum t_pescaador;  
VACUUM  
idx=#  
idx=# █
```

# Vacuum

```
idx=# vacuum full;  
VACUUM  
idx=# 
```



# Vacuum

- Deve monitorar:
  - Linhas mortas (dead rows);
  - Tabela de uso de disco;
  - A ultima execucao (vacuum ou autovacuum);

# Vacuum

```
idx=# SELECT relname, n_dead_tup FROM pg_stat_user_tables;
      relname      | n_dead_tup
-----+-----
 t_pescador_has_t_porto |          0
 t_pescador        |          0
 t_porto           |          0
(3 registros)
```

```
idx=# █
```

# Vacuum

```
idx=# SELECT  relname AS "table_name",
idx-#         pg_size_pretty(pg_table_size(C.oid)) AS "table_size"
idx-# FROM pg_class C
idx-# LEFT JOIN pg_namespace N ON (N.oid = C.relnamespace)
idx-# WHERE nspname NOT IN ('pg_catalog', 'information_schema')
idx-# AND nspname !~ '^pg_toast' AND relkind IN ('r')
idx-# ORDER BY pg_table_size(C.oid)
idx-# DESC LIMIT 1;
 table_name | table_size
-----+-----
 t_pescador | 128 kB
(1 registro)

idx=#
```

# Vacuum

```
idx=# SELECT relname, last_vacuum, last_autovacuum  
idx=# FROM pg_stat_user_tables;
```

relname	last_vacuum	last_autovacuum
t_pescador_has_t_porto		
t_pescador	2018-05-14 14:11:53.287904-03	
t_porto		

(3 registros)

```
idx=#
```

```
idx=# █
```

# Multi-Version Concurrency Control (MVCC)

# Multi-Version Concurrency Control (MVCC)

- Método de controle de concorrência;
  - Garantir as propriedades ACID;

# Ativando LOG para Medir o Tempo da Consulta

```
mohonda@positivo:indices_avancado> psql -d idx
psql (9.6.8)
Digite "help" para ajuda.

idx=# select * from t_pescador;
idx=#
idx=# set log_min_duration_statement = 0;
SET
idx=# SET client_min_messages = log;
LOG:  duração: 0.071 ms  comando: SET client_min_messages= log;
SET
idx=#
idx=# █
```

# Ativando LOG para Medir o Tempo da Consulta

```
mohonda@positivo:indices_avancado> psql -d idx
psql (9.6.8)
Digite "help" para ajuda.

idx=# select * from t_pescador;
idx=#
idx=# set log_min_duration_statement = 0;
SET
idx=# SET client_min_messages = log;
LOG:  duração: 0.071 ms  comando: SET client_min_messages = log;
SET
idx=#
idx=# select * from t_pescador;
LOG:  duração: 1.854 ms  comando: select * from t_pescador;
idx=#
idx=# █
```



# Referências

- KORTH, H. F., SILBERSCHATZ, A. Sistema de Banco de Dados, Makron Books
- DATE, C. J. Introdução a Sistemas de Banco de Dados. Tradução da 7ª. Edição Americana. Editora Campus.
- DATE, C.J. Bancos de Dados, Tópicos Avançados, Editora Campus
- GARCIA-MOLINA, H. ULLMAN, J.D., WIDOM, J. Implementação de Sistemas de Bancos de Dados. Editora Campus
- Manuais Técnicos e Livros voltados para Bancos de Dados específicos.