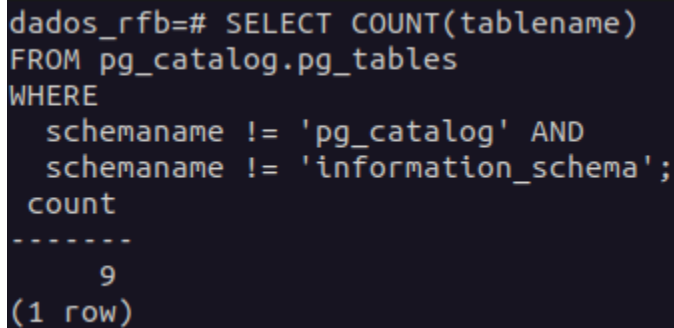


Metadados e estatísticas

- a quantidade de tabelas do banco de dados

```
SELECT COUNT(tablename)
FROM pg_catalog.pg_tables
WHERE
schemaname != 'pg_catalog' AND
schemaname != 'information_schema';
```



```
dados_rfb=# SELECT COUNT(tablename)
FROM pg_catalog.pg_tables
WHERE
schemaname != 'pg_catalog' AND
schemaname != 'information_schema';
count
-----
      9
(1 row)
```

- a quantidade de atributos de cada tabela

```
SELECT table_name, count(column_name)
FROM
information_schema.columns
WHERE table_name IN (
SELECT
table_name
FROM information_schema.tables
WHERE table_schema = 'public'
)
GROUP BY table_name;
```

```

dados_rfb=# SELECT table_name, count(column_name)
FROM
    information_schema.columns
WHERE table_name IN (
    SELECT
        table_name
    FROM information_schema.tables
    WHERE table_schema = 'public'
)
GROUP BY table_name;
  table_name | count
-----+-----
cnae        |      2
empresa     |      7
estabelecimento |     31
moti        |      2
munic       |      2
natju       |      2
pais        |      2
simples     |      7
socios      |     11
(9 rows)

```

- o tamanho de cada tabela em gigabytes

```

SELECT
    table_name,
    pg_size_pretty(pg_relation_size(quote_ident(table_name))),
    pg_relation_size(quote_ident(table_name))
FROM information_schema.tables
WHERE table_schema = 'public'
ORDER BY 3 DESC;

```

```

dados_rfb=# SELECT
  table_name,
  pg_size_pretty(pg_relation_size(quote_ident(table_name))),
  pg_relation_size(quote_ident(table_name))
FROM information_schema.tables
WHERE table_schema = 'public'
ORDER BY 3 DESC;

```

table_name	pg_size_pretty	pg_relation_size
estabelecimento	11 GB	11836030976
empresa	5167 MB	5417525248
simples	2846 MB	2984075264
socios	2586 MB	2711371776
munic	272 kB	278528
cnae	128 kB	131072
pais	16 kB	16384
moti	8192 bytes	8192
natju	8192 bytes	8192

(9 rows)

- a quantidade de acessos sequenciais realizada em cada tabela

```

SELECT relname, seq_scan
FROM pg_stat_user_tables;

```

```

dados_rfb=# SELECT relname, seq_scan
FROM pg_stat_user_tables;

```

relname	seq_scan
natju	0
empresa	21
estabelecimento	36
moti	0
pais	15
simples	21
socios	0
munic	0
cnae	21

(9 rows)

Elaboração das consultas

-- Consulta 1

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento E
INNER JOIN cnae C
ON E.cnae_fiscal_principal = C.codigo
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
GROUP BY 1;
```

-- Consulta 2

```
SELECT E.uf as estado, count(E.cnpj_basico) as quantidade
FROM estabelecimento E
INNER JOIN cnae C ON E.cnae_fiscal_principal = C.codigo
INNER JOIN pais on pais.codigo = E.pais
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
AND pais.descricao = 'BRASIL'
GROUP BY 1
ORDER BY 2 ASC;
```

-- Consulta 3

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
WHERE SI.opcao_pelo_simples = 'S' AND ano_inicio_atividade BETWEEN '2015' AND '2023'
GROUP BY 1;
```

-- Consulta 4

```
SELECT ES.uf as estado, count(ES.cnpj_basico) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN pais on pais.codigo = ES.pais
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
AND pais.descricao = 'BRASIL'
GROUP BY 1
ORDER BY 2 ASC;
```

-- Consulta 5

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
WHERE SI.opcao_mei = 'S' AND ano_inicio_atividade BETWEEN '2015' AND '2023'
GROUP BY 1;
```

-- Consulta 6

```
SELECT ES.uf as estado, count(ES.cnpj_basico) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN pais on pais.codigo = ES.pais
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
      AND pais.descricao = 'BRASIL'
GROUP BY 1
ORDER BY 2 ASC;
```

-- Consulta 7

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento E
INNER JOIN cnae C
ON E.cnae_fiscal_principal = C.codigo
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
      AND C.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100')
GROUP BY 1
ORDER BY 1;
```

-- Consulta 8

```
SELECT E.uf as estado, count(E.cnpj_basico) as quantidade
FROM estabelecimento E
INNER JOIN cnae C ON E.cnae_fiscal_principal = C.codigo
INNER JOIN pais on pais.codigo = E.pais
WHERE ano_inicio_atividade BETWEEN '2015' AND '2023'
      AND C.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100')
      AND pais.descricao = 'BRASIL'
GROUP BY 1
ORDER BY 2 ASC;
```

-- Consulta 9

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN cnae CN ON ES.cnae_fiscal_principal = CN.codigo
WHERE SI.opcao_pelo_simples = 'S'
      AND ano_inicio_atividade BETWEEN '2015' AND '2023'
      AND CN.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100')
GROUP BY 1
ORDER BY 1;
```

-- Consulta 10

```
SELECT ES.uf as estado, count(ES.cnpj_basico) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN cnae CN ON ES.cnae_fiscal_principal = CN.codigo
INNER JOIN pais on pais.codigo = ES.pais
WHERE SI.opcao_pelo_simples = 'S'
AND ano_inicio_atividade BETWEEN '2015' AND '2023'
AND CN.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100')
AND pais.descricao = 'BRASIL'
GROUP BY 1
ORDER BY 2 ASC;
```

-- Consulta 11

```
SELECT ano_inicio_atividade as ano, count(data_inicio_atividade) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN cnae CN ON ES.cnae_fiscal_principal = CN.codigo
WHERE SI.opcao_mei = 'S'
AND ano_inicio_atividade BETWEEN '2015' AND '2023'
AND CN.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100',
'9511800')
GROUP BY 1;
```

-- Consulta 12

```
SELECT ES.uf as estado, count(ES.cnpj_basico) as quantidade
FROM estabelecimento ES
INNER JOIN empresa EM ON ES.cnpj_basico = EM.cnpj_basico
INNER JOIN simples SI ON SI.cnpj_basico = EM.cnpj_basico
INNER JOIN cnae CN ON ES.cnae_fiscal_principal = CN.codigo
WHERE SI.opcao_mei = 'S'
AND ano_inicio_atividade BETWEEN '2015' AND '2023'
AND CN.codigo IN ('6201500', '6201501', '6202300', '6203100', '6204000', '6209100',
'9511800')
GROUP BY 1
ORDER BY 2 ASC;
```

Avaliação das consultas

1) Com o buffer limpo execute as consultas sem qualquer método de acesso disponível.

2) Atualize as estatísticas do banco de dados.

3) Execute cada uma das consultas definidas sem a utilização de índice e obtenha o tempo de execução. Qual o tempo obtido? Que consultas demoraram mais? E quais demoraram menos e a que você atribui esse desempenho?

Tempo da consulta 1: 30362,901 ms (00:30,363)
Tempo da consulta 2: 28678,482 ms (00:28,678)
Tempo da consulta 3: 56487,182 ms (00:56,487)
Tempo da consulta 4: 71659,650 ms (01:11,660)
Tempo da consulta 5: 88586,847 ms (01:28,587)
Tempo da consulta 6: 129527,030 ms (02:09,527)
Tempo da consulta 7: 22323,785 ms (00:22,324)
Tempo da consulta 8: 4227,020 ms (00:04,227)
Tempo da consulta 9: 36023,165 ms (00:36,023)
Tempo da consulta 10: 43321,917 ms (00:43,322)
Tempo da consulta 11: 48567,508 ms (00:48,568)
Tempo da consulta 12: 37698,155 ms (00:37,698)

4) Para cada consulta, sugira um índice não clusterizado, execute a consulta com o índice limpando o buffer do banco de dados. Verifique se o banco de dados utiliza o índice (verifique com o comando explain). Obtenha o tempo de execução das consultas. Qual o tempo obtido? Que consultas demoraram mais? E quais demoraram menos e a que você atribui esse desempenho?

Indices:

```
CREATE INDEX empresa_cnpj_basico_hash_index ON empresa USING HASH(cnpj_basico);  
CREATE INDEX simples_cnpj_basico_hash_index ON simples USING HASH(cnpj_basico);  
CREATE INDEX estabelecimento_cnpj_basico_hash_index ON estabelecimento USING  
HASH(cnpj_basico);  
CREATE INDEX cnae_codigo_hash_index ON cnae USING HASH(codigo);  
CREATE INDEX ano_inicio_atividade_estabelecimento_btree_index ON estabelecimento  
USING BTREE(ano_inicio_atividade);
```

Consulta 1:

Indices usados:

Tempo de execução: 10775,006 ms (00:10,775)

Consulta 2:

Indices usados:

Tempo de execução: 5103,722 ms (00:05,104)

Consulta 3:

Índices usados:

Tempo de execução: 83134,453 ms (01:23,134)

Consulta 4:

Índices usados:

using empresa_cnpj_basico_hash_index

Tempo de execução: 53843,991 ms (00:53,844)

Consulta 5:

Índices usados:

Tempo de execução: 102624,478 ms (01:42,624)

Consulta 6:

Índices usados:

using empresa_cnpj_basico_hash_index

using simples_cnpj_basico_hash_index

Tempo de execução: 64337,307 ms (01:04,337)

Consulta 7:

Índices usados:

cnae_codigo_hash_index

Tempo de execução: 18917,265 ms (00:18,917)

Consulta 8:

Índices usados:

cnae_codigo_hash_index

Tempo de execução: 4638,862 ms (00:04,639)

Consulta 9:

Índices usados:

cnae_codigo_hash_index

simples_cnpj_basico_hash_index

empresa_cnpj_basico_hash_index

Tempo de execução: 35596,157 ms (00:35,596)

Consulta 10:

Índices usados:

cnae_codigo_hash_index

simples_cnpj_basico_hash_index

empresa_cnpj_basico_hash_index

Tempo de execução: 13736,030 ms (00:13,736)

Consulta 11:

Índices usados:

cnae_codigo_hash_index

simples_cnpj_basico_hash_index

empresa_cnpj_basico_hash_index

Tempo de execução: 67104,460 ms (01:07,104)

Consulta 12:

Índices usados:

cnae_codigo_hash_index
simples_cnpj_basico_hash_index
empresa_cnpj_basico_hash_index

Tempo de execução: 69968,822 ms (01:09,969)

5) Para qual consulta você sugeriria um índice clusterizado e porquê? Crie o índice clusterizado e re-execute a consulta. Verifique se o SGBD utilizou o índice e colete o tempo de execução.

Cluster:

CLUSTER estabelecimento USING ano_inicio_atividade_estabelecimento_btree_index;

Consulta 1:

Índices usados:

Tempo de execução: 6569,904 ms (00:06,570)

Consulta 2:

Índices usados:

Tempo de execução: 4261,316 ms (00:04,261)

Consulta 3:

Índices usados:

Tempo de execução: 115271,419 ms (01:55,271)

Consulta 4:

Índices usados:

empresa_cnpj_basico_hash_index
simples_cnpj_basico_hash_index

Tempo de execução: 48331,951 ms (00:48,332)

Consulta 5:

Índices usados:

Tempo de execução: 0130,261 ms (01:30,130)

Consulta 6:

Índices usados:

empresa_cnpj_basico_hash_index
simples_cnpj_basico_hash_index

Tempo de execução: 48252,162 ms (00:48,252)

Consulta 7:

Índices usados:

cnae_codigo_hash_index

Tempo de execução: 23398,573 ms (00:23,399)

Consulta 8:

Índices usados:

cnae_codigo_hash_index

Tempo de execução: 8706,782 ms (00:08,707)

Consulta 9:

Índices usados:

cnae_codigo_hash_index
simples_cnpj_basico_hash_index
empresa_cnpj_basico_hash_index

Tempo de execução: 40125,430 ms (00:40,125)

Consulta 10:

Índices usados:

cnae_codigo_hash_index
simples_cnpj_basico_hash_index
empresa_cnpj_basico_hash_index

Tempo de execução: 17636,567 ms (00:17,637)

Consulta 11:

Índices usados:

cnae_codigo_hash_index
simples_cnpj_basico_hash_index
empresa_cnpj_basico_hash_index

Tempo de execução: 53811,939 ms (00:53,812)

Consulta 12:

Índices usados:

cnae_codigo_hash_index
simples_cnpj_basico_hash_index
empresa_cnpj_basico_hash_index

Tempo de execução: 68004,746 ms (01:08,005)