



Como detectar e corrigir índices corrompidos

- Arquivo do índice está corrompido no disco. Causas :
 - ✓ fsync ou full_page_writes = off + crash
 - ✓ bug do fsync do sistema operacional
 - ✓ pentes de memória sem ECC com defeito

```
flavio=> \d tbl_demo
          Table "public.tbl_demo"
  Column   | Type   | Modifiers
-----+-----+-----
 id        | bigint | not null default nextval('tbl_demo_id_seq'::regclass)
 name_hash | text   |
Indexes:
    "tbl_demo_pkey" PRIMARY KEY, btree (id)

flavio=> EXPLAIN (analyze) SELECT * FROM tbl_demo ORDER BY id;
ERROR:  could not read block 1280 in file "base/16385/16578": read only 0 of 8192 bytes
```

```
flavio=> SELECT relname, relkind FROM pg_class WHERE relfilenode = 16578;
 relname    | relkind
-----+-----
 tbl_demo_pkey | i
(1 row)
```

- Identificar a causa
 - corrigir a configuração, atualizar o sistema, testar RAM e hardware em geral.
- REINDEX
 - Com um pouco de sorte, testar e migrar para um servidor standby.

```
flavio=> reindex index tbl_demo_pkey ;
REINDEX
flavio=> EXPLAIN (analyze) SELECT * FROM tbl_demo ORDER BY id;
                                QUERY PLAN
-----
Index Scan using tbl_demo_pkey on tbl_demo  (cost=0.42..35329.43 rows=1000000 width=41) (actual time=0.040..131.839 rows=1000000 loops=1)
  Planning time: 0.228 ms
  Execution time: 155.092 ms
(3 rows)
```

- Causada por um bug no PostgreSQL até as versões:
 - 9.2.19
 - 9.3.15
 - 9.4.10
 - 9.5.5
 - 9.6.1
- CREATE INDEX CONCURRENTLY
- Quando tem UPDATE concorrente
- O índice criado pode ter falta de algumas linhas da tabela

- Consultas podem mostrar resultados diferentes, se elas usam o índice corrompido
- Chaves únicas ou primárias podem não ser respeitadas
- Chaves estrangeiras podem não ser respeitadas
- Uma solução rápida é necessária

- Utilizar uma transação em modo REPEATABLE READ ou SERIALIZABLE
 - Alternativamente, com o banco sem escrita, no modo READ COMMITTED
- Forçar um SELECT com seqscan
- Forçar um outro SELECT com o índice
- Comparar os resultados

- Cansado? Centenas de índices a testar? Tabelas enormes? Não tem certeza?
 - **pg_check_index.rb** faz o trabalho tedioso!
 - https://github.com/doctolib/pg_check_indexes
- Escrito em Ruby
 - Ruby 2.4.0 ou superior
 - Gem pg
 - Gem optparse
- Pode testar todo o banco, somente uma tabela ou tabelas até um tamanho indicado.
 - Testado com PostgreSQL 9.4, 9.5 et 9.6

- Testa todos os índices do tipo btree
 - integridade
 - unicidade (inclusive chaves primárias e restrições únicas)
 - com funções
 - colunas múltiplas
 - parciais (com cláusula WHERE)
- Testa todas as chaves estrangeiras
- Não testa outros tipos de índice
- Não modifica nada (não destrutivo)
- Não corrige eventuais corrupções
- Pode ser executado sobre um servidor standby (réplica física, por streaming ou

```
$ ruby pg_check_indexes.rb
*** START OF INDEX/TABLE CONSISTENCY TEST ***

Table: public.tbl_demo - Index: tbl_demo_pkey - Columns: id, comparing data... index and table match
- OK. Index is unique or a PK, testing uniqueness... no duplicate entries - OK.
Table: public.unicorn_timings - Index: unicorn_timings_server_idx - Columns: server, comparing
data... index and table match - OK.
Table: public.unicorn_timings - Index: unicorn_timings_tmst_idx - Columns: tmst, comparing data...
index and table match - OK.

*** END OF INDEX/TABLE CONSISTENCY TEST ***
*** START OF FOREIGN KEYS CONSISTENCY TEST ***

Table: public.tbl_fk_demo - Constraint: tbl_fk_demo_tbl_demo_id_fkey - Columns: tbl_demo_id -
Referenced table: tbl_demo - Columns: id. Checking... no missing entries in referenced table - OK.
*** END OF FOREIGN KEYS CONSISTENCY TEST ***
```

- **Atualizar PostgreSQL !!!**
- REINDEX imediato se possível
 - Escrita não é possível durante REINDEX
 - As consultas podem mostrar resultados diferentes após REINDEX
- Executar novamente `pg_check_index.rb`
 - Olhar principalmente chaves estrangeiras que podem se “corrigir” após o REINDEX

■ Alternativa

- CREATE (UNIQUE) INDEX CONCURRENTLY ...;
- DROP INDEX ...;
- ALTER INDEX ... RENAME TO ...;

■ Se chave primária

- CREATE UNIQUE INDEX CONCURRENTLY ...;
- ALTER TABLE ... DROP CONSTRAINT ...;
- ALTER TABLE ... ADD CONSTRAINT ... PRIMARY KEY USING INDEX ...;

Obrigado!

- www.doctolib.fr

Estamos recrutando

- www.doctolib.fr/jobs