Timbira

inteligência_em_postgres

Tuning de Consulta

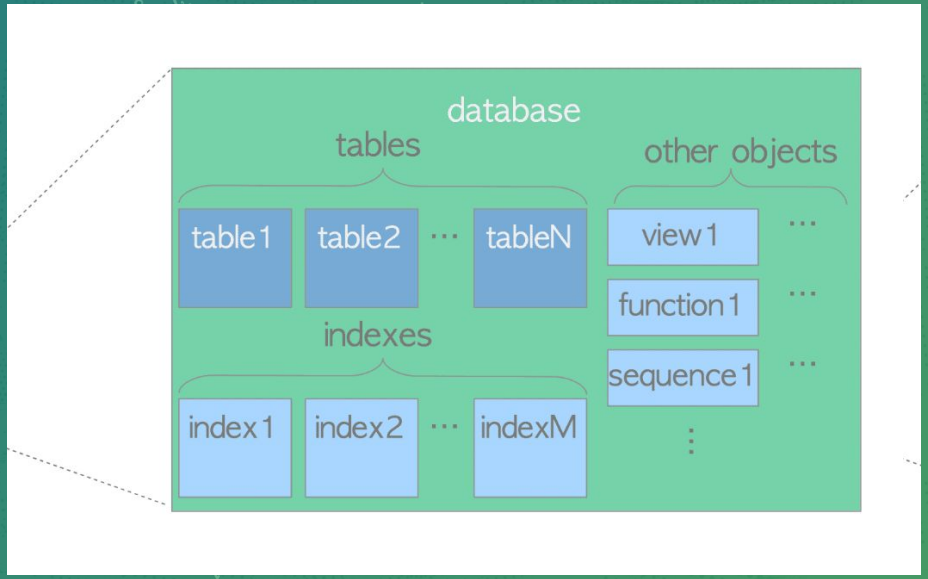
AGENDA

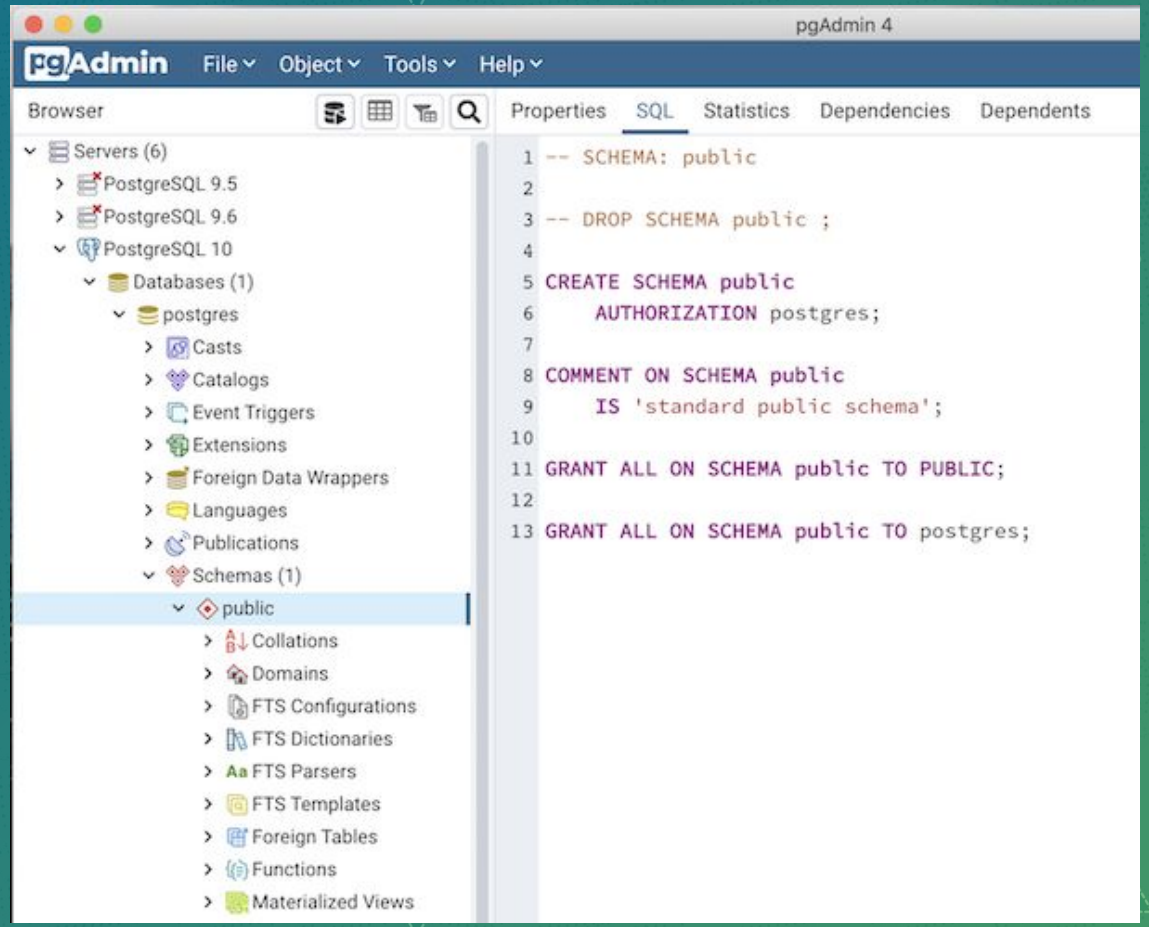
- Objetivos
- Objetos no banco de dados
- Use índice
- Índice nem sempre resolve tudo
- Data types e funções

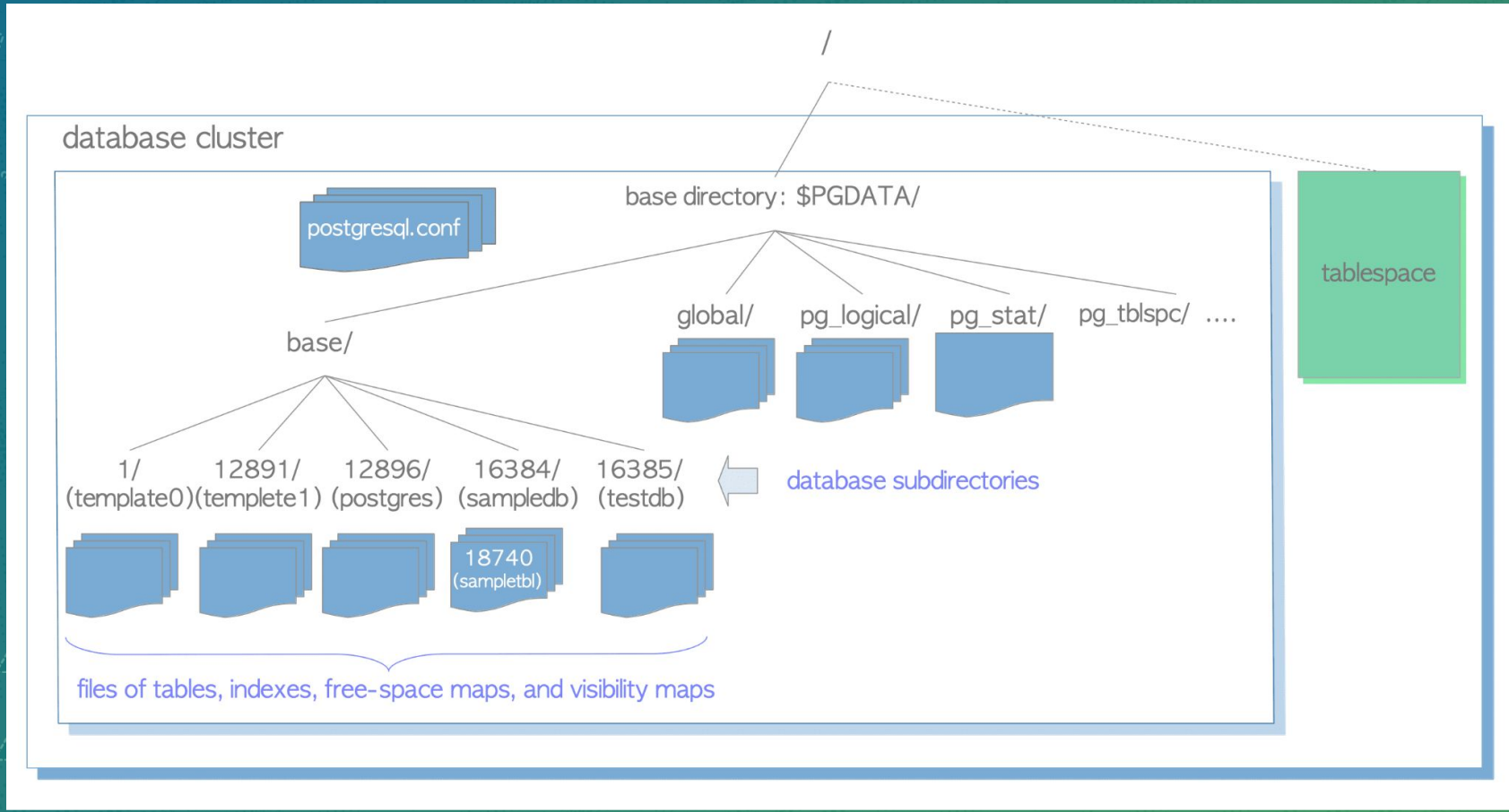
Objetivos

- Nosso primeiro objetivo é subir a régua;
- Passar conhecimento para prevenir erros assim produzindo softwares melhores

Objetos no banco de dados



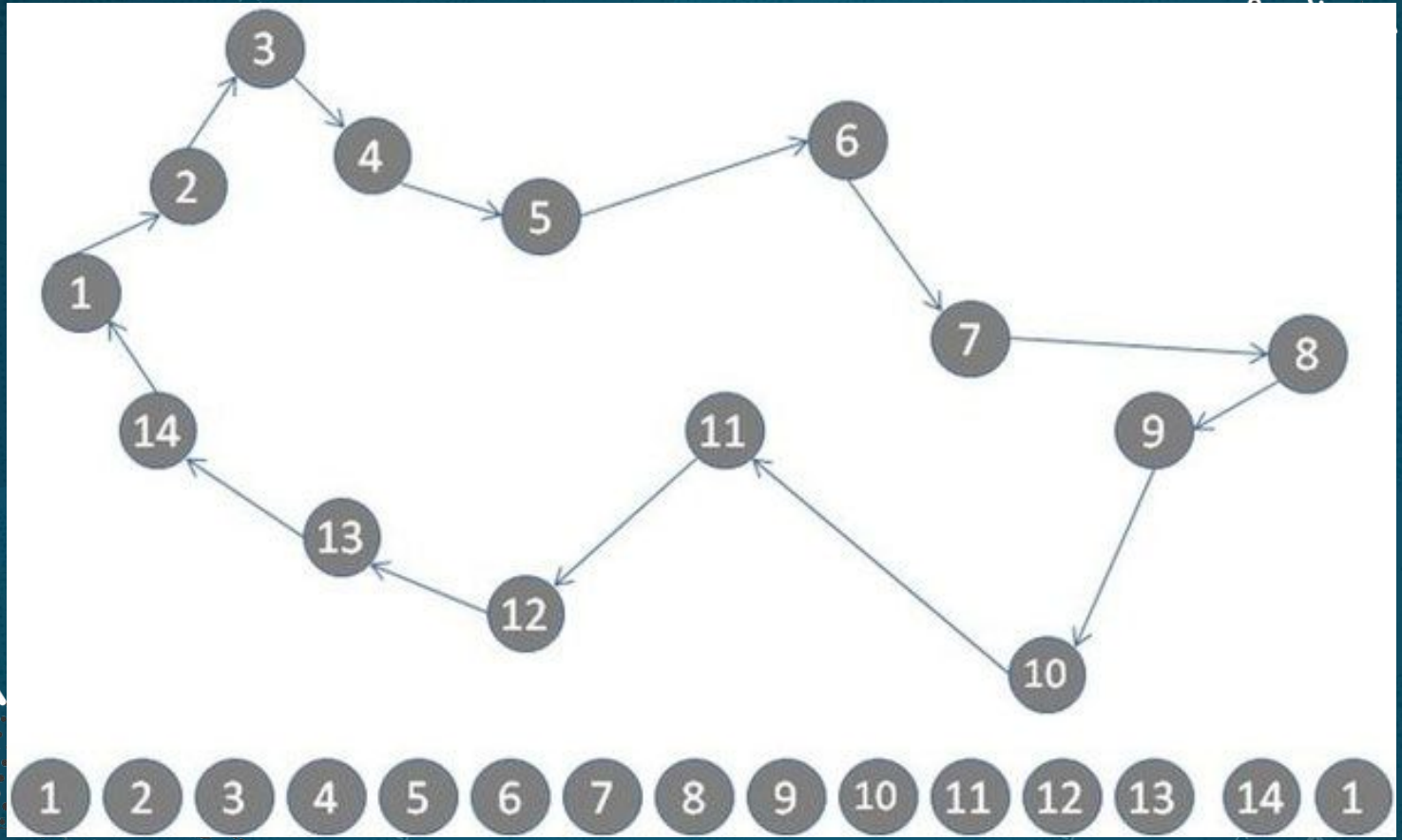

pgAdmin 4
pgAdmin
File
Object
Tools
Help
Browser
Properties
SQL
Statistics
Dependencies
Dependents
Servers (6)
PostgreSQL 9.5
PostgreSQL 9.6
PostgreSQL 10
Databases (1)
postgres
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Publications
Schemas (1)
public
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
1 -- SCHEMA: public
2
3 -- DROP SCHEMA public ;
4
5 CREATE SCHEMA public
6 AUTHORIZATION postgres;
7
8 COMMENT ON SCHEMA public
9 IS 'standard public schema';
10
11 GRANT ALL ON SCHEMA public TO PUBLIC;
12
13 GRANT ALL ON SCHEMA public TO postgres;


/
database cluster
base directory: $PGDATA/
postgresql.conf
tablespace
base/
global/
pg_logical/
pg_stat/
pg_tblspc/
....
1/
(template0)
12891/
(templete1)
12896/
(postgres)
16384/
(sampledb)
16385/
(testdb)
database subdirectories
18740
(sampletbl)
files of tables, indexes, free-space maps, and visibility maps

Falar sobre performance não é fácil

Existem muitos aspectos envolvidos.

- Hardware
- SO e filesystems
- Configuração
- Tipo de carga
- Consultas


3
4
6
2
5
1
7
8
14
11
9
13
12
10
1 2 3 4 5 6 7 8 9 10 11 12 13 14 1

Como resolver uma consulta lenta?

Vamos aos exemplos !

Métricas e Observalidade

O sistema tá lento !

Não dá para conversar sobre o tal sistema está lento se não houver Métricas e Observalidade.

Exemplo 1

Um sistema com que busca no banco de dados uma quantidade de pessoas e exibe em uma tela.

-DDL
CREATE TABLE PESSOA (ID BIGINT , CODIGO VARCHAR);
CREATE TABLE

-LOAD
INSERT INTO PESSOA (SELECT GENERATE_SERIES,MD5(GENERATE_SERIES::VARCHAR) FROM GENERATE_SERIES(1,100000));
INSERT 0 100000

Consulta

```
SELECT * FROM PESSOA_PROD WHERE ID >=10 AND ID<100;
```

Métricas e Observalidade

O básico.

TEMPO !

```
timbira=# \timing
Timing is on.
```

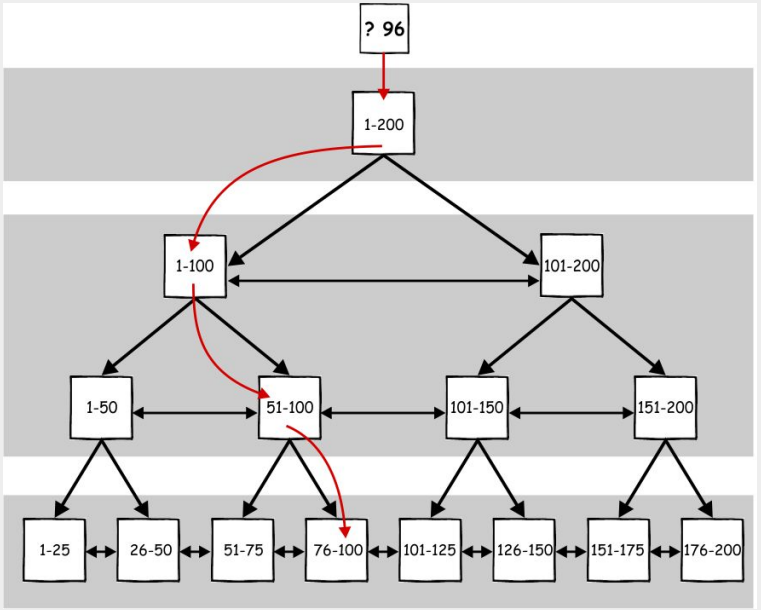
Média de : 10.975 ms

Criando um índice

-DDL
CREATE INDEX pessoa_id_idx ON PESSOA (ID);

-CONSULTA
SELECT * FROM PESSOA WHERE ID >=10 AND ID<100;

Média: 0.785 ms


? 96
1-200
1-100
101-200
1-50
51-100
101-150
151-200
1-25
26-50
51-75
76-100
101-125
126-150
151-175
176-200

Use Índice !

- É muito eficiente
- Parece mágico
- Mas não resolve todos os problemas

Exemplo 2

—DDL
CREATE TABLE PESSOA (ID BIGINT **PRIMARY KEY**, CODIGO VARCHAR);

—LOAD
INSERT INTO PESSOA (SELECT GENERATE_SERIES,MD5(GENERATE_SERIES::VARCHAR) FROM GENERATE_SERIES(1,1000000));

Consulta

SELECT * FROM PESSOA WHERE ID >=10 AND ID<=1000;

Média: 0.701 ms

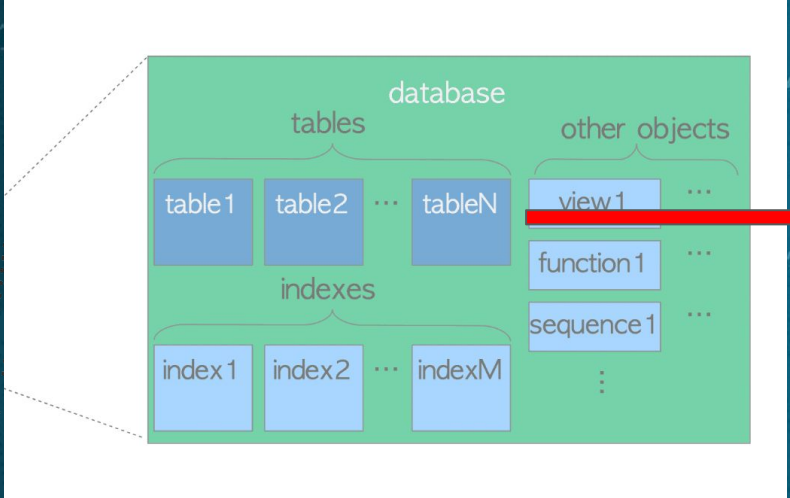
Em produção a história é outra

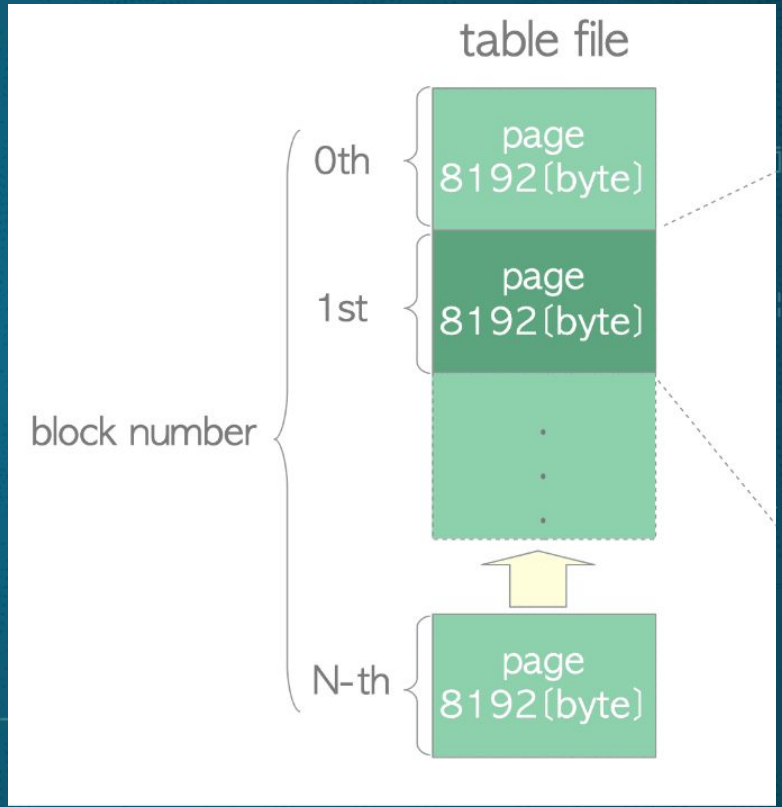
Média: 4.663 ms

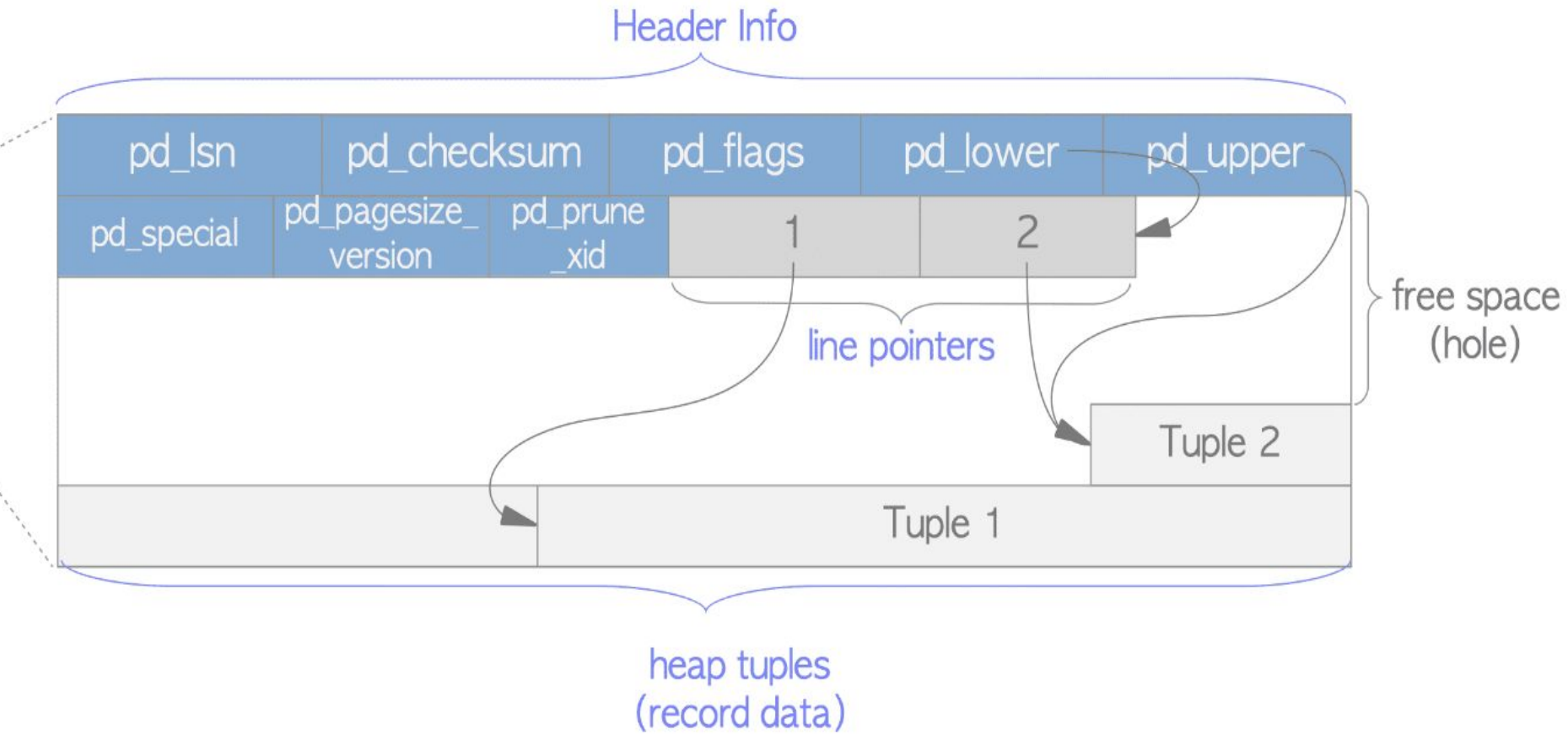
Mas os dados não estão iguais ?

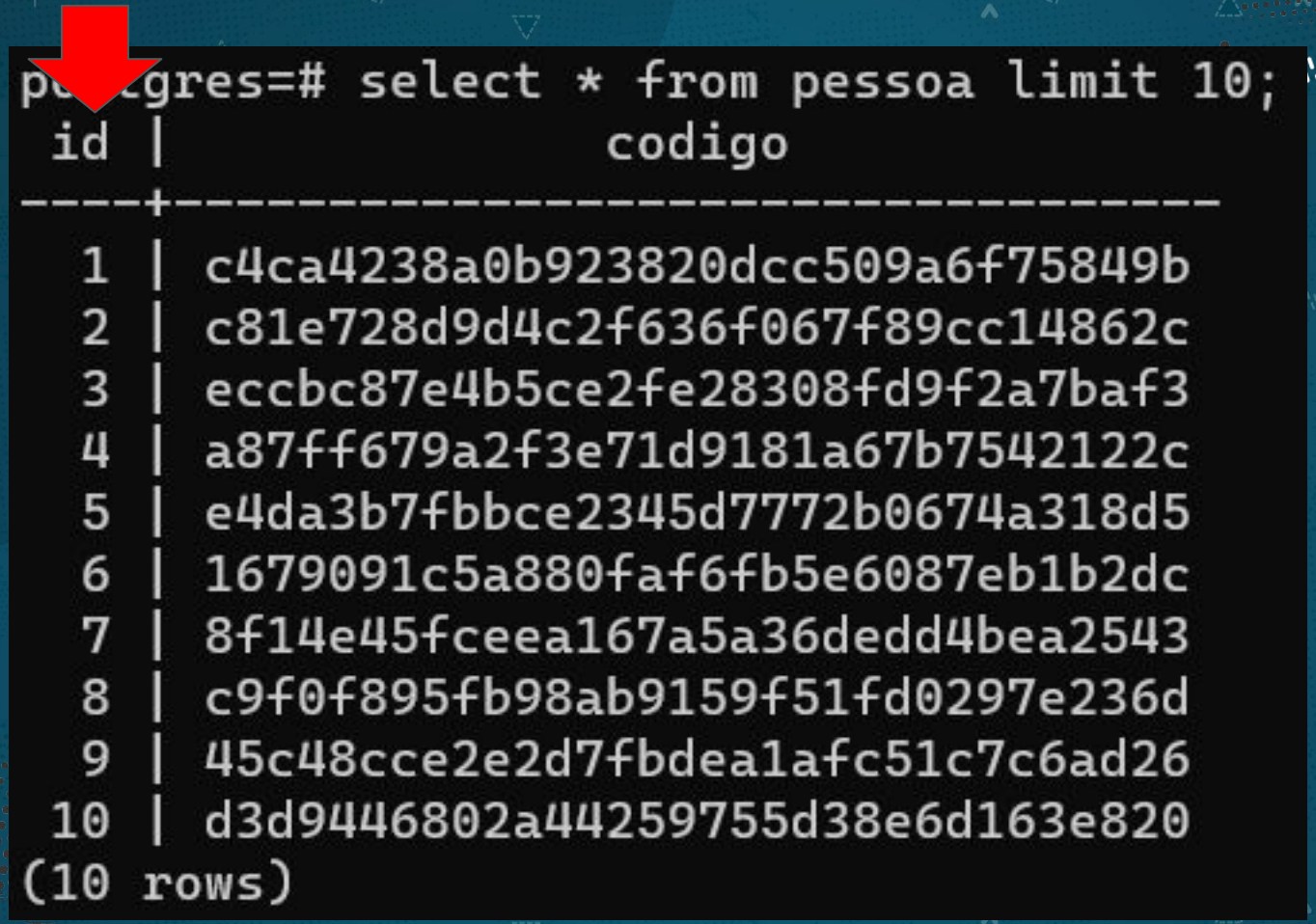
```
                                List of relations
 Schema |    Name     | Type  |  Owner   | Persistence | Access method | Size  | Description
--------+-------------+-------+----------+-------------+---------------+-------+-------------
 public | pessoa      | table | postgres | permanent   | heap          | 73 MB |
 public | pessoa_prod | table | postgres | permanent   | heap          | 73 MB |
(2 rows)
```

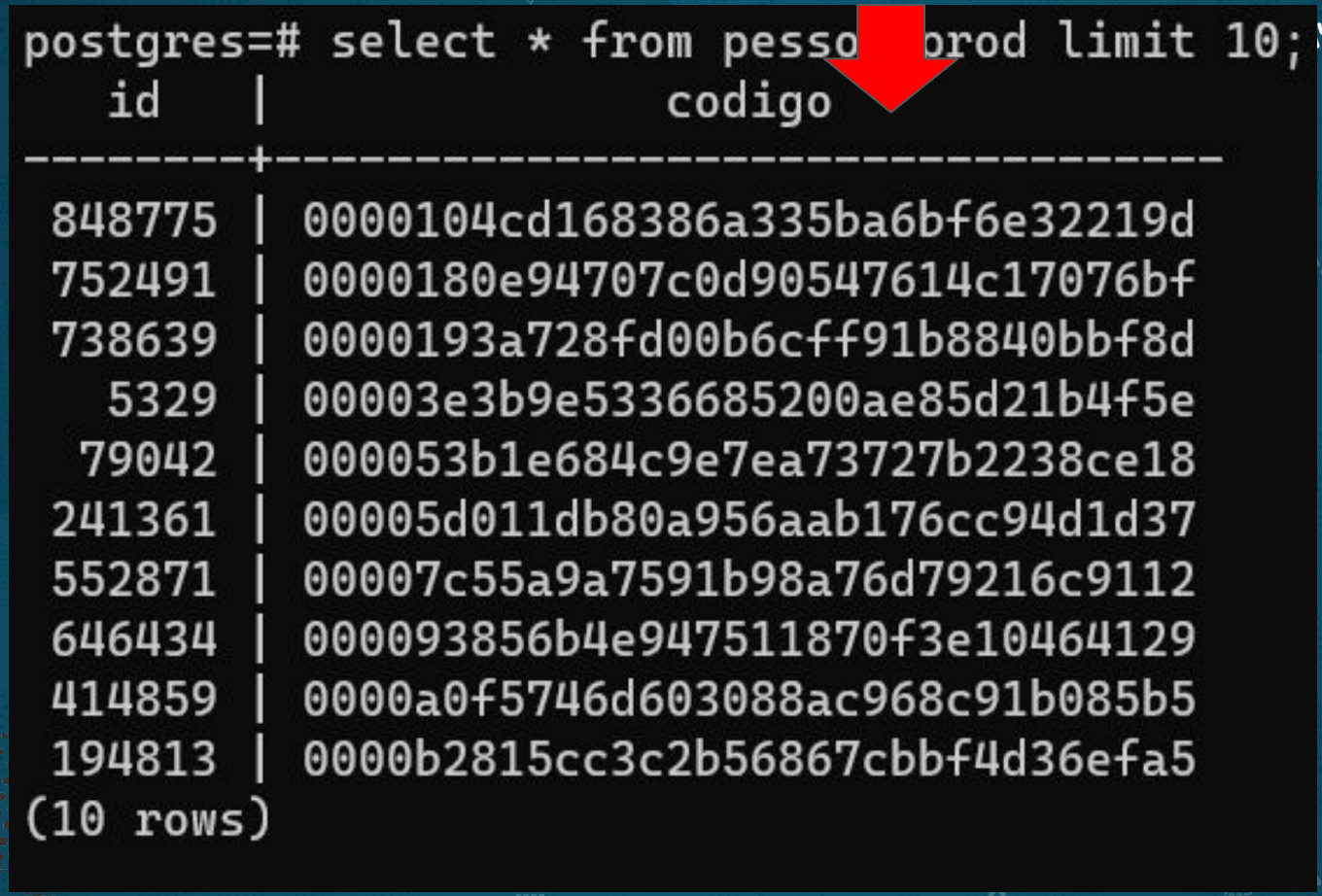
Linhas x Páginas



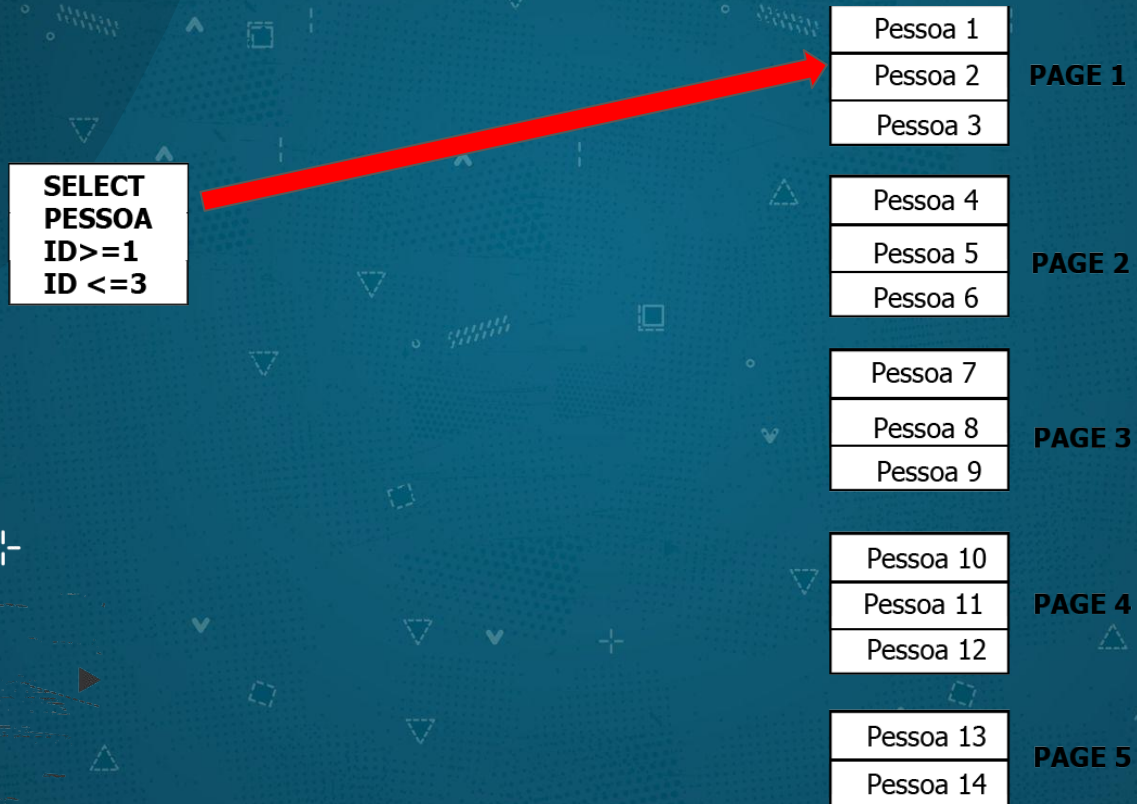



Header Info
pd_lsn
pd_checksum
pd_flags
pd_lower
pd_upper
pd_special
pd_pagesize_
version
pd_prune
_xid
1
2
line pointers
free space
(hole)
Tuple 2
Tuple 1
heap tuples
(record data)

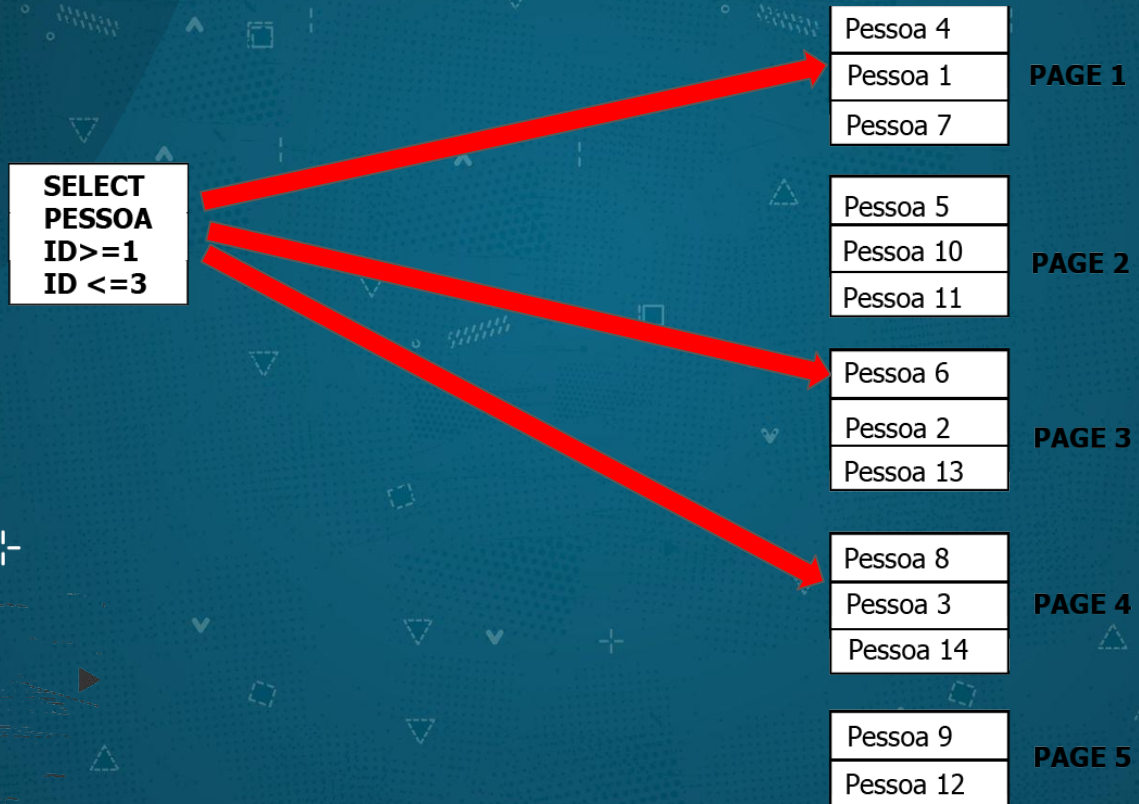

postgres=# select * from pessoa limit 10;
id | codigo
1 | c4ca4238a0b923820dcc509a6f75849b
2 | c81e728d9d4c2f636f067f89cc14862c
3 | eccbc87e4b5ce2fe28308fd9f2a7baf3
4 | a87ff679a2f3e71d9181a67b7542122c
5 | e4da3b7fbbce2345d7772b0674a318d5
6 | 1679091c5a880faf6fb5e6087eb1b2dc
7 | 8f14e45fceea167a5a36dedd4bea2543
8 | c9f0f895fb98ab9159f51fd0297e236d
9 | 45c48cce2e2d7fbdea1afc51c7c6ad26
10 | d3d9446802a44259755d38e6d163e820
(10 rows)


postgres=# select * from pesso prod limit 10;
id | codigo
848775 | 0000104cd168386a335ba6bf6e32219d
752491 | 0000180e94707c0d90547614c17076bf
738639 | 0000193a728fd00b6cff91b8840bbf8d
5329 | 00003e3b9e5336685200ae85d21b4f5e
79042 | 000053b1e684c9e7ea73727b2238ce18
241361 | 00005d011db80a956aab176cc94d1d37
552871 | 00007c55a9a7591b98a76d79216c9112
646434 | 000093856b4e947511870f3e10464129
414859 | 0000a0f5746d603088ac968c91b085b5
194813 | 0000b2815cc3c2b56867cbbf4d36efa5
(10 rows)

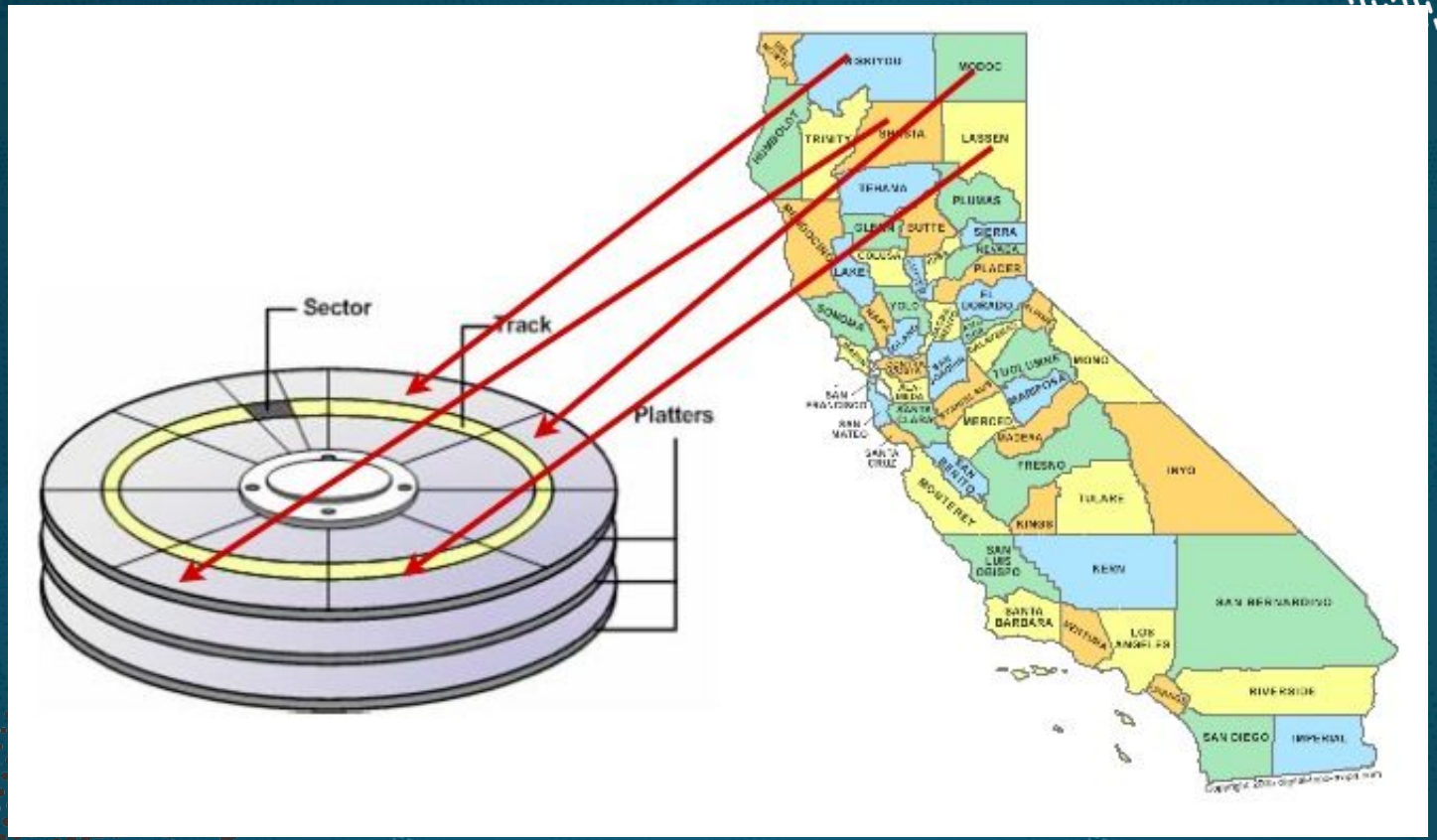
Banco de dados ler páginas

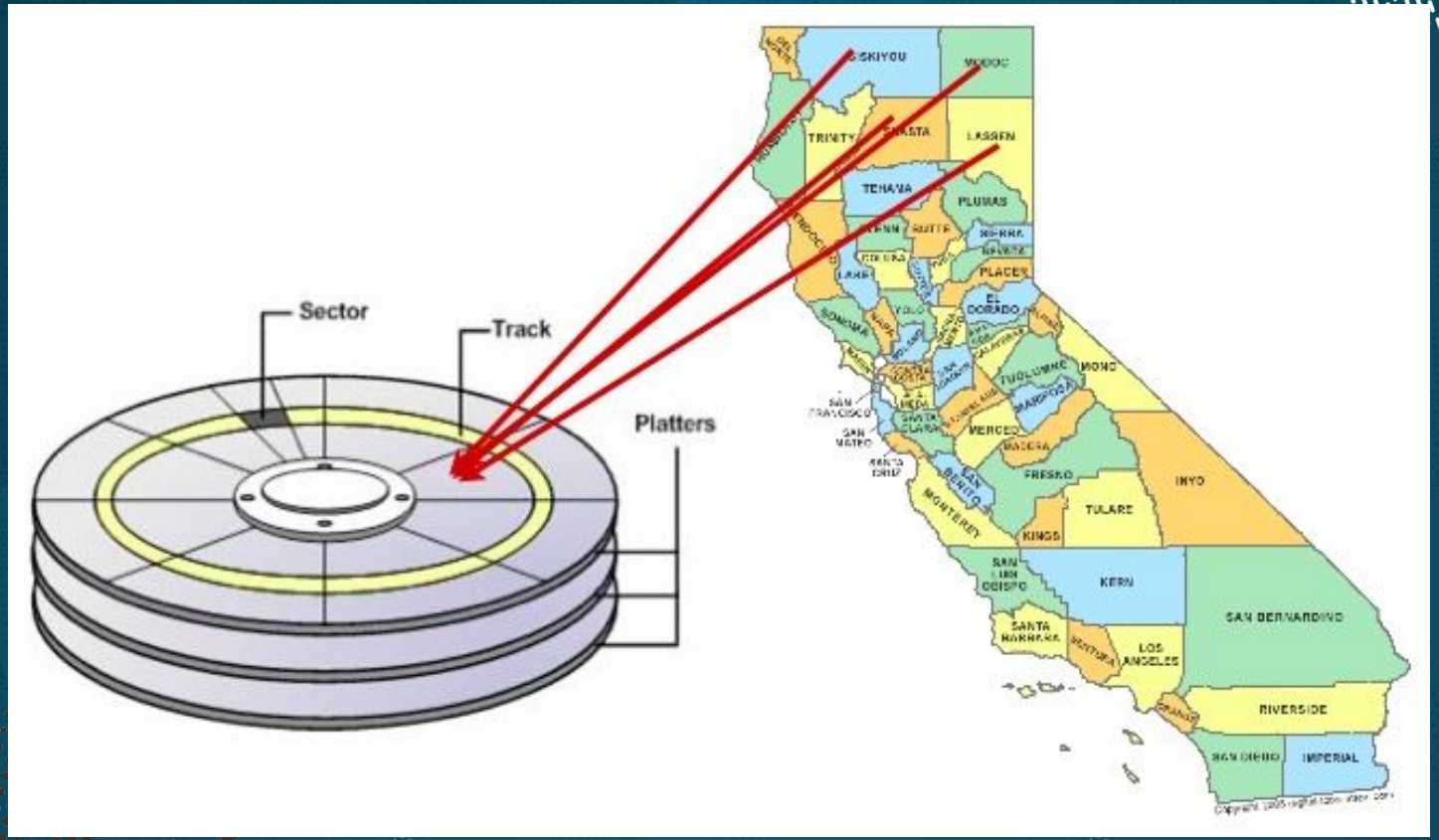


Banco de dados ler páginas



CLUSTER pessoa_prod USING pessoa_prod_pkey ;

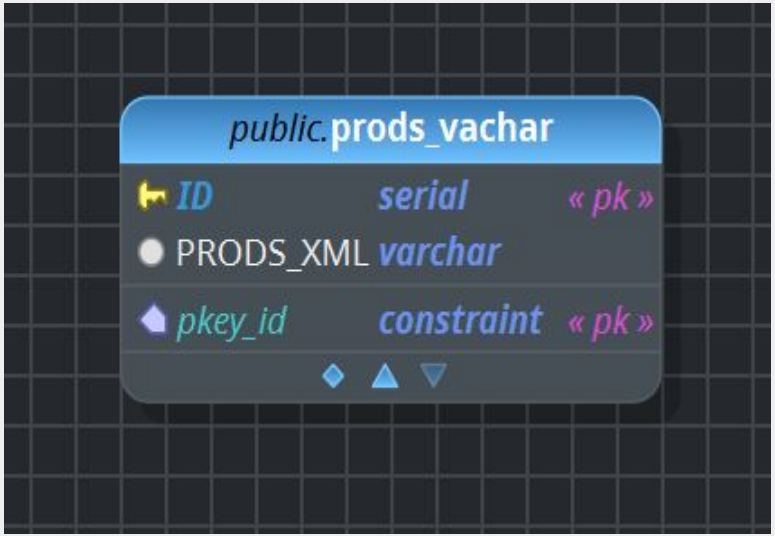

Sector
Track
Platters
MODOC
LASSEN
PLUMAS
SIERRA
PLACER
EL DORADO
TUOLUMNE
MARIPOSA
MERCED
MADERA
FRESNO
TULARE
INYO
KINGS
KERN
SAN BERNARDINO
RIVERSIDE
SAN DIEGO
IMPERIAL
SAN LUIS OBISPO
SANTA BARBARA
VENTURA
LOS ANGELES
ORANGE
MONTEREY
SAN BENITO
SANTA CRUZ
SAN MATEO
SAN FRANCISCO
SONOMA
LAKE
YOLO
SUTTER
YUBA
NEVADA
TRINITY
HUMBOLDT
TEHAMA
SHASTA


Sector
Track
Platters
SISKIYOU
MODOC
TRINITY
SHASTA
LASSEN
TEHAMA
PLUMAS
SUTTER
PLACER
EL DORADO
MONO
MARIPOSA
MERCED
MADERA
FRESNO
INYO
TULARE
KINGS
KERN
SAN BERNARDINO
LOS ANGELES
RIVERSIDE
IMPERIAL

Data types e funções

- Numeric Types
- Monetary Types
- Character Types
- Binary Data Types
- UUID Type
- Date/Time Types
- Boolean Type
- Geometric Types
- XML Type
- JSON Types

Exemplo 3




```
—DDL
CREATE TABLE PRODS_VARCHAR (ID SERIAL
PRIMARY KEY, PROD_XML VARCHAR);

—LOAD
\i prods_202306141713.sql
```

XML

```xml
<det nItem="1">
    <prod>
        <cProd>70697</cProd>
        <cEAN>SEM GTIN</cEAN>
        <xProd>prod Um</xProd>
        <NCM>69090</NCM>
        <CEST>0000000</CEST>
        <CFOP>5102</CFOP>
        <uCom>UN</uCom>
        <qCom>1.0000</qCom>
        <vUnCom>26.9000000000</vUnCom>
        <vProd>26.90</vProd>
        <cEANTrib>SEM GTIN</cEANTrib>
        <uTrib>UN</uTrib>
        <qTrib>1.0000</qTrib>
        <vUnTrib>26.9000000000</vUnTrib>
        <indTot>1</indTot>
    </prod>
    <imposto>
        <vTotTrib>0.00</vTotTrib>
        <ICMS>
            <ICMS00>
                <orig>0</orig>
```

Consulta

```
SELECT

REGEXP_REPLACE(prod_xml ,'(.*)(<det nItem=)(.*)(><prod>)(.*)','\3') as
numeroprod,

REGEXP_REPLACE(prod_xml ,'(.*)(<qCom>)(.*)(</qCom>)(.*)','\3') as qCom,

REGEXP_REPLACE(prod_xml ,'(.*)(<vProd>)(.*)(</vProd>)(.*)','\3') as vProd

FROM public.prods_varchar WHERE ID =2;
```

Tempo: 1.745 ms

—DDL
CREATE TABLE PRODS_XML (ID SERIAL PRIMARY KEY, PROD_XML **XML**);

—LOAD
\i prods_202306141713.sql

Consulta

```
SELECT

XPATH('/det/@nItem',prod_xml)  numeroprod,

XPATH('/det/prod/qCom/text()',prod_xml) qCom,

XPATH('/det/prod/vProd/text()',prod_xml) vProd from
public.prods_xml WHERE ID =2;
```

Tempo: 0.581 ms

- Use índice
- Índice nem sempre resolve tudo
- Banco de dados ler páginas
- Use o tipo de dados correto

luisfelipe0805

eu.luis.felipe

CURTIU?
SEGUE A GENTE!

/TimbiraBrasil
/Timbira

timbira.com.br