



PGConf.Brasil 2018

Postgres no Governo Federal: Ponto de Vista do Desenvolvedor

Leonardo Leite
Serpro

Eu

- Engenheiro de computação (Poli USP)
- Mestre em Ciência da Computação (IME USP)
- Desenvolvedor de sistemas no **Serpro**
- Doutorando em Ciência da Computação (IME USP)



Agradecimentos

- Altair Lima
- Jorge Rossetto
- Aline Hatadani
- Rodolfo Bottossi
- Roberta Monteiro
- Juliano Atanazio

Fatiador

<https://gitlab.com/serpro/fatiador/>



Esta palestra...

PostgreSQL - SQL Básico

Juliano Atanazio

1

de 402

PostgreSQL - SQL Básico



Juliano Atanazio

Esta palestra

- Compartilhar alguns conteúdos avançados que realmente são utilizados em meu trabalho!
- Perspectiva do desenvolvedor.

Dimensões

- 850 GB
- 1,7 TB
- 3M reg / dia
- 15 TB

Q

Particionamento

```
registro_auditoria  
registro_auditoria_01_2014  
registro_auditoria_02_2014  
registro_auditoria_03_2014  
...
```

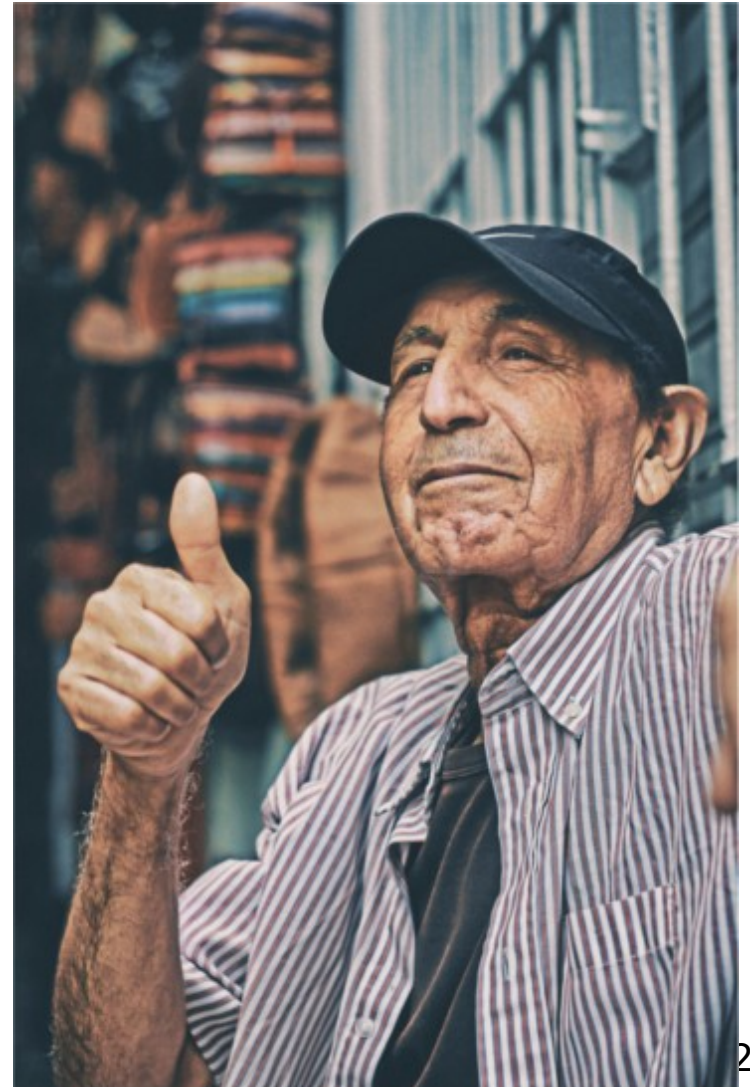
Particionamento

Q



Particionamento

Obs: PostgreSQL 10 +1



Hibernate

JPQL vs SQL

DAOs...

- Interface baseada no negócio.
- Interface conveniente para o negócio.
- Otimizar acesso ao banco.
 - Tentar fazer uma query só.

Uma query só... em JPA

- Às vezes muito difícil em JPQL!
- JPQL não tem nem count xD

DAOs (no JPA)

- Não hesite em trocar o JPQL por SQL quando necessário / conveniente
- Dica: necessário / conveniência em geral = evitar **for's**.

Exemplo

```
"select
csv.identificacao_csv, csv.data_inspecao, csv.itl_responsavel_nome_engenheiro,
csv.itl_responsavel_cpf_engenheiro, csv.itl_responsavel_nome, csv.itl_responsavel_cnpj,
inspecao.data_registro, inspecao.uf_itl, csv.aprovado, csv.proprietario_documento,
veiculo_inspeccionado.placa, veiculo_inspeccionado.chassi, csv.tipo_csv,
string_agg(csv_escopo_inspeccionado.nome, ' | ') as nomes_escopos_inspecao,
string_agg(csv_escopo_inspeccionado.autorizacao_numero, ' | ') as numeros_autorizacoes_previas,
estado_csv.estado, veiculo_inspeccionado.tipo
from
siscsv.csv, siscsv.estado_csv, siscsv.inspecao,
siscsv.csv_veiculo as veiculo_inspeccionado, siscsv.csv_escopo_inspeccionado
where
csv.identificacao_csv = inspecao.identificacao_csv and
csv.id_veiculo_inspeccionado = veiculo_inspeccionado.id and
csv.identificacao_csv = csv_escopo_inspeccionado.identificacao_csv
and csv.identificacao_csv = estado_csv.identificacao_csv and estado_csv.estado in "
+ filter.getEstadosCsvPermitidosAsQueryParameter());
```

Aliás...

```
string_agg(csv_escopo_inspecionado.nome, ' | ')  
as nomes_escopos_inspecao
```

string_agg



string_agg

```
select
  pessoa.nome as nome_pessoa,
  cachorro.nome as nome_cachorro,
from
  pessoa, cachorro
where
  pessoa.id = cachorro.id_pessoa
```

nome_pessoa	nome_cachorro
Roger Dearly	Pongo
Roger Dearly	Perdy
Dick Vigarista	Muttley

string_agg

```
select
  pessoa.nome as nome_pessoa,
  string_agg(cachorro.nome, ', ' ) as nomes_cachorros,
from
  pessoa, cachorro
where
  pessoa.id = cachorro.id_pessoa
```

nome_pessoa	nomes_cachorros
Roger Dearly	Pongo, Perdy
Dick Vigarista	Muttley

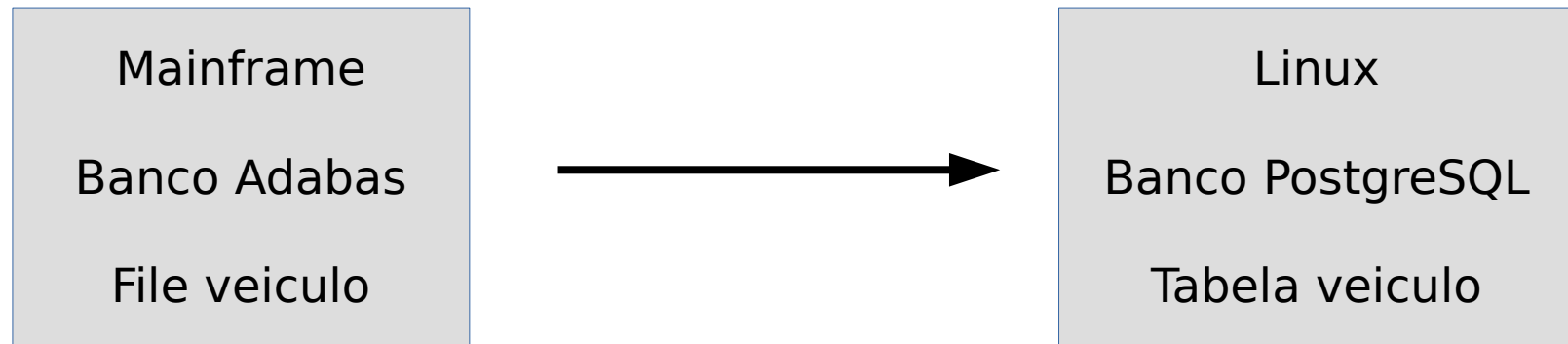
Outro exemplo útil para relatórios

```
select count(case when aprovado = true then 1 end)  
as aprovados from csv;
```

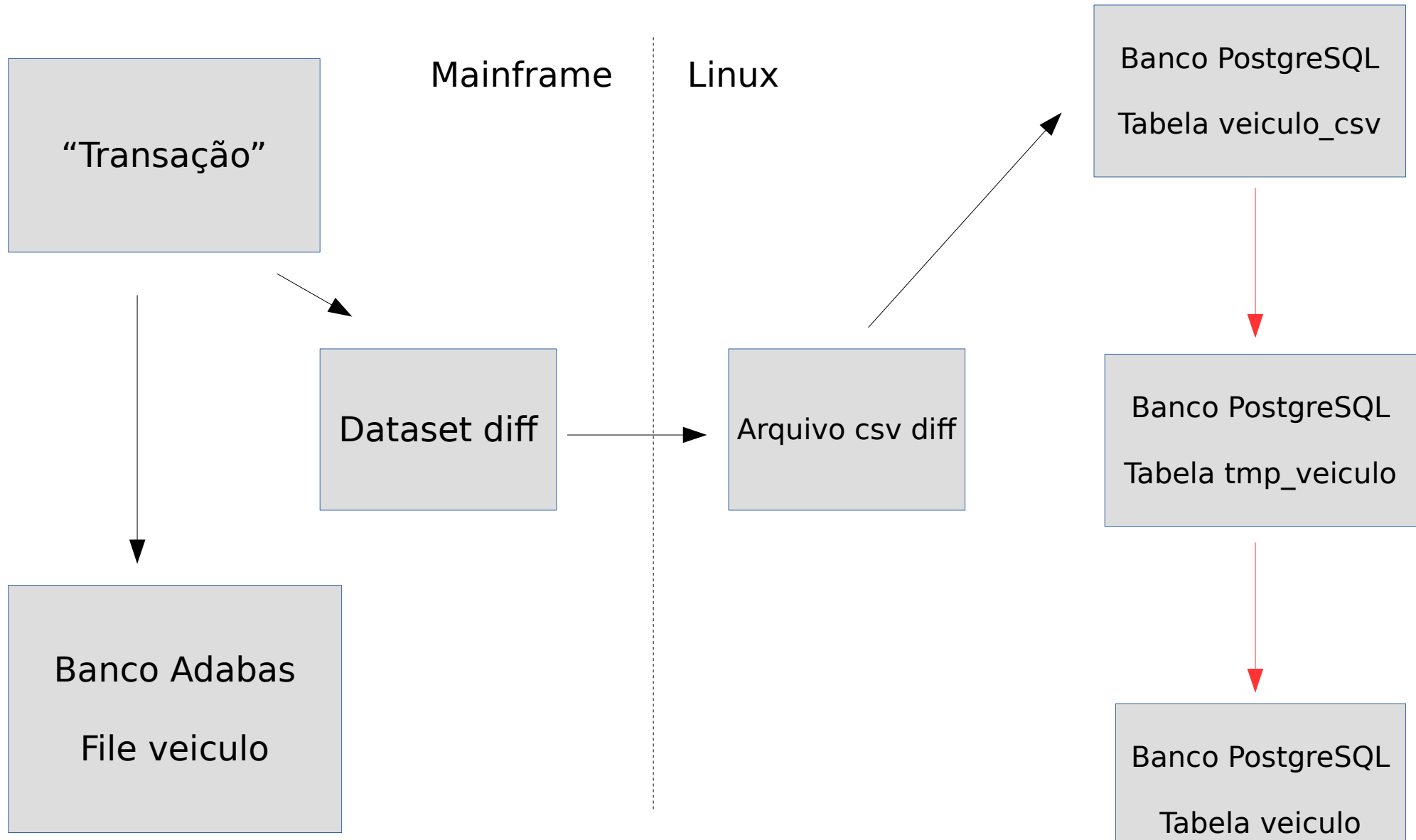
```
select count(case when doente = true then 1 end)  
as quantidade_de_cachorros_doentes from cachorro;
```

```
select sum(1) filter (where doente = true)  
as quantidade_de_cachorros_doentes from cachorro;
```

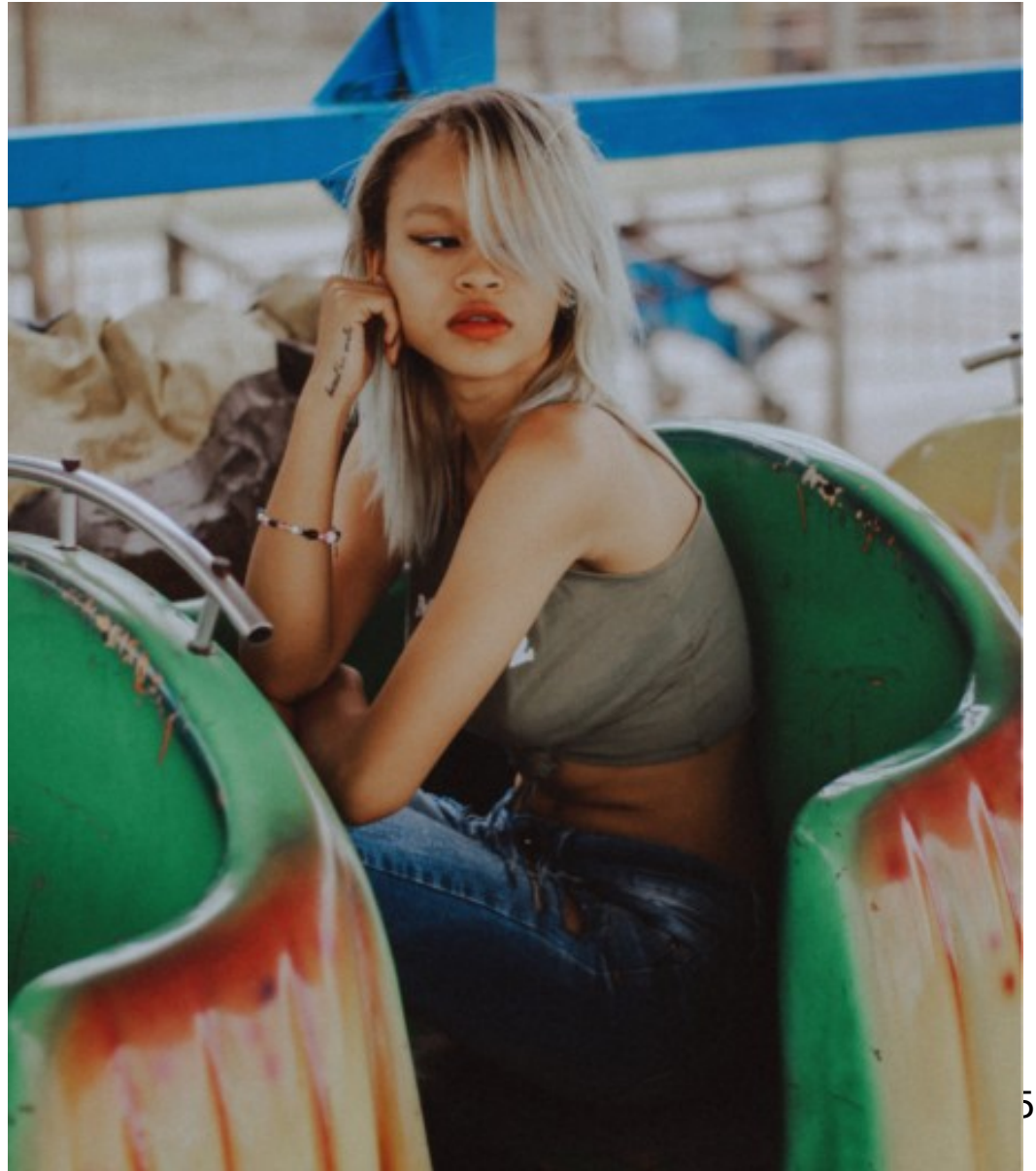
Replicação entre bancos



Replicação entre bancos



Triggers



Q

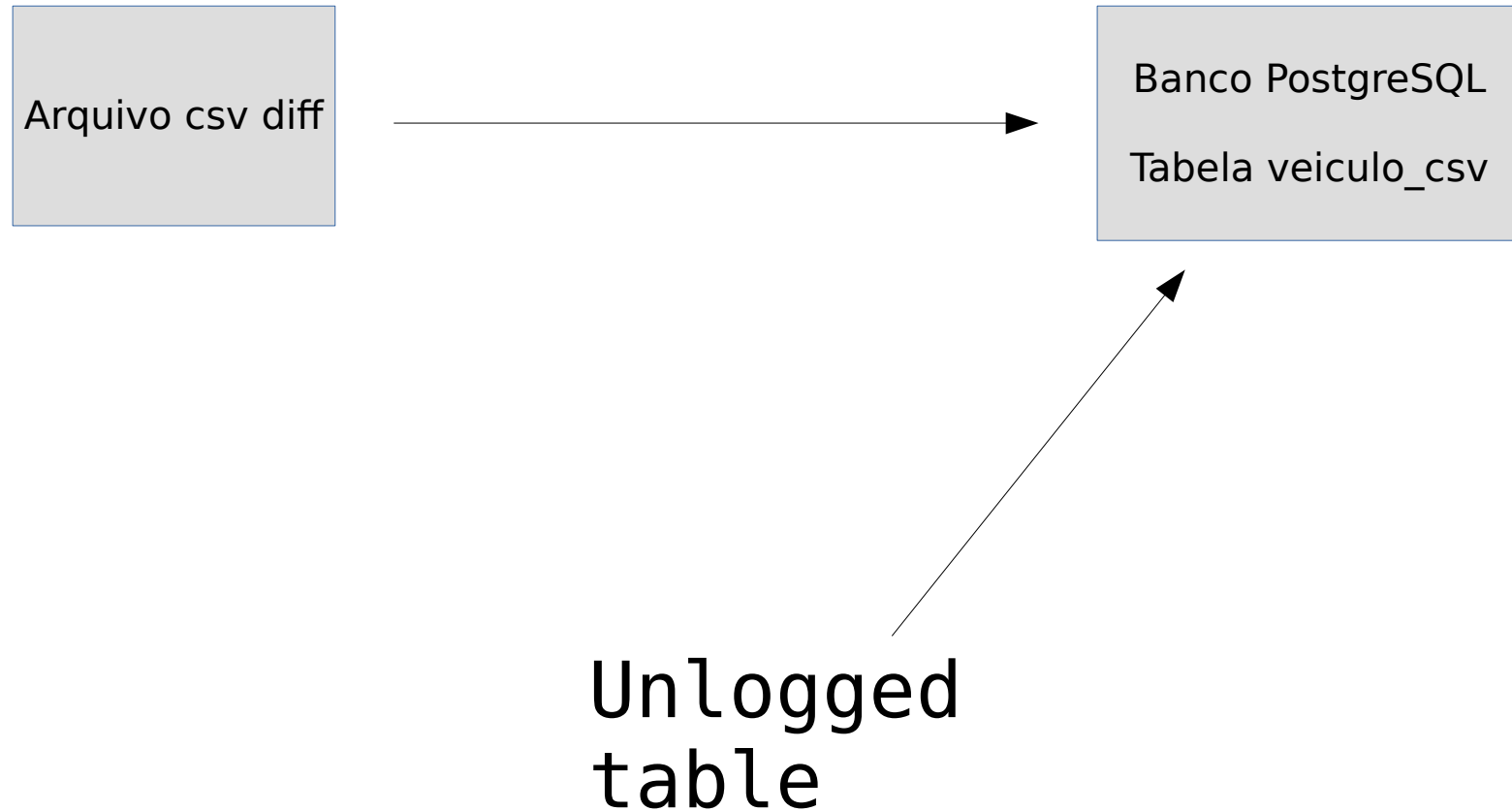
FDW - foreign-data wrapper



```
CREATE EXTENSION file_fdw;
CREATE SERVER local_server FOREIGN DATA WRAPPER file_fdw;

CREATE FOREIGN TABLE IF NOT EXISTS veiculo_csv (
    identificacao_unica    text,
    renavam                text,
    placa                  text,
    chassi                 text,
    data_hora_atualizacao text
    ...
) SERVER local_server
OPTIONS ( filename '/opt/veiculos.csv', format 'csv',
         delimiter ';', escape ';', quote '"');
```

FDW - foreign-data wrapper

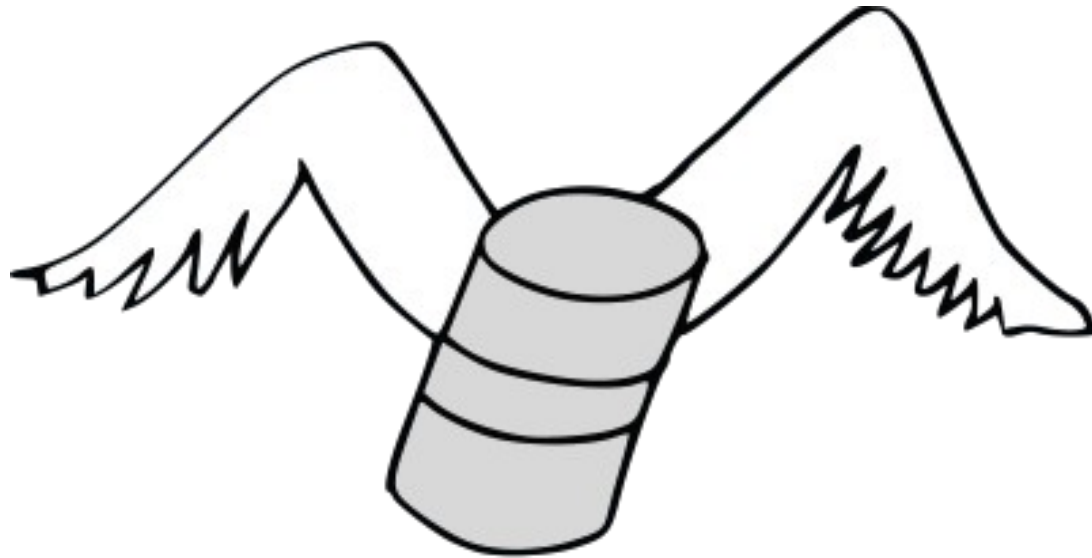


(FDW x BaaS)

Upsert / partition over

```
insert into veiculo
with t as (
    select *,
        max(data_hora_atualizacao) over (partition by identificacao_unica) as max_dt
    from tmp_veiculo
)
select
    identificacao_unica,
    renavam,
    placa,
    chassi,
    case when data_hora_atualizacao::numeric = 0 then null
        else to_timestamp(data_hora_atualizacao, 'YYYYMMDDhh24missms') end,
    ...
FROM
    t
WHERE
    data_hora_atualizacao = max_dt
ON CONFLICT (identificacao_unica) DO UPDATE SET
    identificacao_unica = EXCLUDED.identificacao_unica,
    renavam = EXCLUDED.renavam,
    placa = EXCLUDED.placa,
    chassi = EXCLUDED.chassi,
    data_hora_atualizacao = EXCLUDED.data_hora_atualizacao,
    ...;
```

Flyway



al	workspaces	siscsv	SISCSV-Implementacao	01-Applicacao	siscsv-ws	src	main	resources	db	migration
figs										
Nome										
	V0001__estrutura.sql									
	V0002__alterar_dono_tabela.sql									
	V0003__dados_municipios.sql									
	V0004__auditoria_atividade_web.sql									
	V0005__auditoria_requisicao_ws.sql									
	V0006__dados_siscsv.sql									
	V0007__estrutura_complementar.sql									
	V0008__altera_dono_estrutura_complementar.sql									
	V0009__recurso_autorizar_itl_escopo_inmetro.sql									
	V0010__veiculo_atual_para_veiculo_inspecionado.sql									
	V0011__itl_socio.sql									
	V0012__alter_varchar_to_text.sql									
	V0013__recurso_csv_legacy.sql									
	V0014__perfil_detran_master.sql									
	V0015__alter_varchar_to_text_parte_2.sql									
	V0016__alter_codigo_alteracao_permitida_to_text.sql									
	V0017__remove_colunas_inspecao.sql									
	V0018__permissao_alteracao_alteracao_to_alteracao_previa.sql									
	V0019__alter_serial_to_64_digito.sql									
	V0020__escopo_inspecao_exige_indicador_media_monta.sql									
	V0021__grupo_denatran_many_to_many_to_escopo_inspecao.sql									
	V0022__grupo_inmetro_many_to_many_to_escopo_inspecao.sql									
	V0023__libera_relatorios_para_inmetro.sql									
	V0024__alterar_Coluna_identificacao_certificado_para_identificacao_csv.sql									

PL/PGSQL

Bom evitar, mas...

Q

PL/PGSQL

```
DO $$  
DECLARE  
user_ VARCHAR;  
BEGIN  
IF current_user <> 'postgres' THEN  
    user_ := replace(current_user, 'user', 'owner');  
    EXECUTE 'ALTER TABLE qrcode OWNER TO ' || user_;  
END IF;  
END $$
```

Reassign

reassign owned by old_role to new_role

Testes com bancos em memória

Q

Testes com bancos em memória

#treta

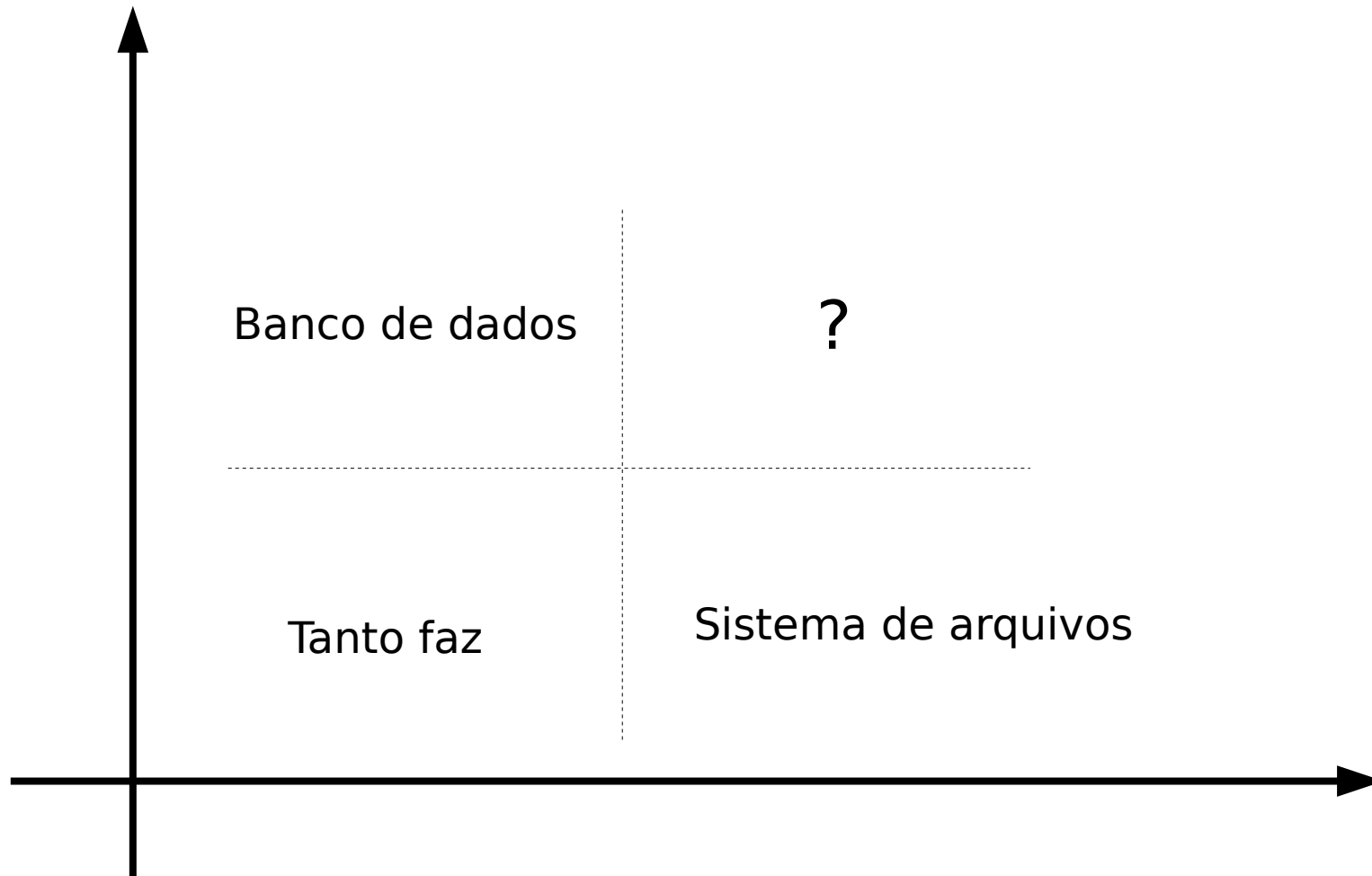
Testes com bancos em memória

- Coisas que não funfaram:
 - SQL nativa!!!
 - Chave primária tipo text.
 - Busca por texto acentuado.

Dados binários no banco?

Q

Dados binários



Dados binários

- Usar tipo **bytea**.
- Não usar tipo **blob**.
- Tabela separada

cachorro

id
nome
id_dono
raça
idade

foto_cachorro

id
id_cachorro
foto **bytea**



Dados binários

- Segregados em um **tablespace** a parte
 - Degradação suave
 - Ex: disco lotado
 - Questões de I/O de disco
- Segregados em um **schema** a parte
 - Facilita tarefas administrativas (ex: vacuum)



CEPH

- Distributed
- No single point of failure
- Scalable
- Replicates
- Fault-tolerant
- Minimize administration time
- Disaster recovery

NoSQL?



NoSQL?



Exemplo: amigos no Face.

Q: banco No-SQL para
o cenário da replicação.

Extra

- varchar(n) x text
- CPF: integer vs text



PGConf.Brasil 2018

Leonardo Alexandre Ferreira Leite

leofl@protonmail.com

<https://www.ime.usp.br/~leofl/>

Meu blog sobre SQL:

<https://entrevistandodados.wordpress.com/>