Do backup ao Standby

Como Salvar o dia, o emprego e a empresa

Diamante

AGGRANDIZE

COMMVAULT

TD SYNNEX

Platina

GOLDENGATEBR
Consultoria & Treinamentos

DISCOVER

Ouro

scansource

VERTICA
by opentext

Prata

TRACES

CAFÉ COM CLOUD

ROX
We take care of your data

Apoio

FIAP

GRUPO POSEIDON DIGITAL

Índice

Instalação do PostgreSQL

Carga de dados

pg_basebackup

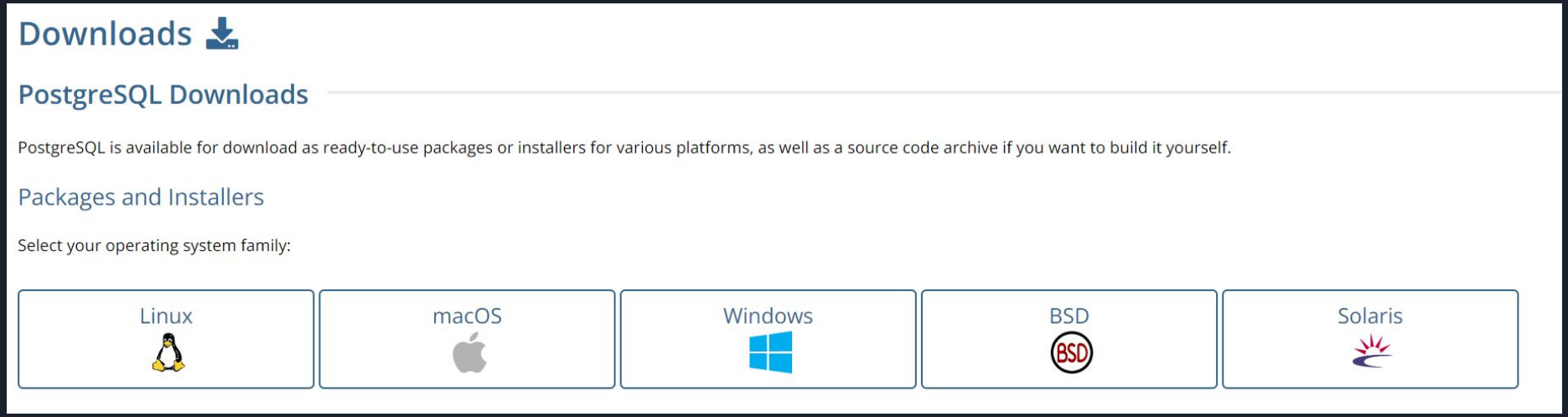
Archives

Point In Time Recovery

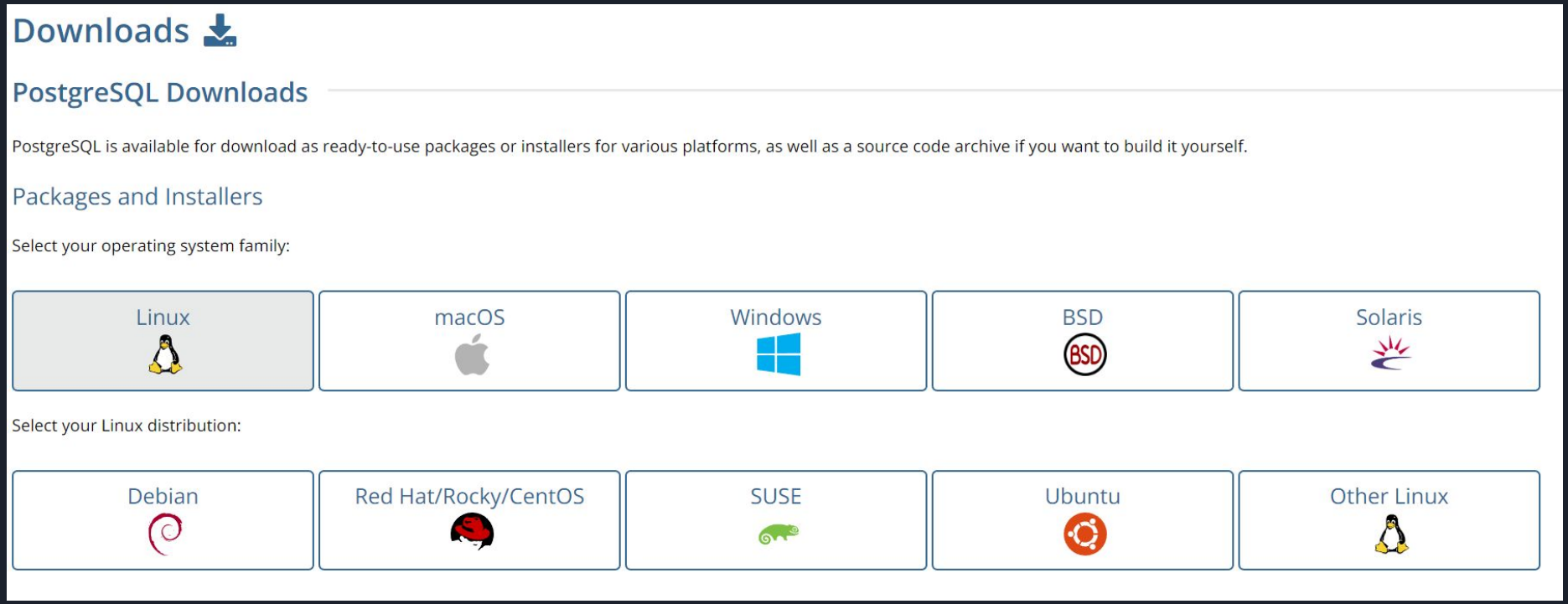
Standby

Talk is cheap
Show me the code!

Instalação


Downloads
PostgreSQL Downloads
PostgreSQL is available for download as ready-to-use packages or installers for various platforms, as well as a source code archive if you want to build it yourself.
Packages and Installers
Select your operating system family:
Linux
macOS
Windows
BSD
Solaris

Instalação


Downloads
PostgreSQL Downloads
PostgreSQL is available for download as ready-to-use packages or installers for various platforms, as well as a source code archive if you want to build it yourself.
Packages and Installers
Select your operating system family:
Linux
macOS
Windows
BSD
Solaris
Select your Linux distribution:
Debian
Red Hat/Rocky/CentOS
SUSE
Ubuntu
Other Linux

Instalação

Linux downloads (Debian)

PostgreSQL is available in all Debian versions by default. However, the stable versions of Debians "snapshot" a specific version of PostgreSQL that is then supported throughout the lifetime of that Debian version. The PostgreSQL project also maintains an apt repository with all supported of PostgreSQL available.

PostgreSQL Apt Repository

If the version included in your version of Debian is not the one you want, you can use the PostgreSQL Apt Repository. This repository will integrate with your normal systems and patch management, and provide automatic updates for all supported versions of PostgreSQL throughout the support lifetime of PostgreSQL.

The PostgreSQL apt repository supports the currently supported stable versions of Debian:

- bookworm (12.x)
- bullseye (11.x)
- buster (10.x)
- sid (unstable)

on the following architectures:

- amd64
- arm64
- i386 (buster and older)
- ppc64el

To use the apt repository, follow these steps:

Copy Script

```
# Create the file repository configuration:
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

# Import the repository signing key:
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -

# Update the package lists:
sudo apt-get update

# Install the latest version of PostgreSQL.
# If you want a specific version, use 'postgresql-12' or similar instead of 'postgresql':
sudo apt-get -y install postgresql
```

Instalação

Na linha de comando:

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt
$(lsb_release -cs)-pgdg main" >
/etc/apt/sources.list.d/pgdg.list'

wget --quiet -O -
https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo
apt-key add -

sudo apt-get update

sudo apt-get -y install postgresql
```

Pós instalação

- Editar /etc/hosts
- Trocar chaves SSH
- Criar um local para guardar o backup

Carga de dados

```
wget https://ftp.postgresql.org/pub/projects/pgFoundry/dbsamples/pagila/pagila/pagila-0.10.1.zip
wget https://ftp.postgresql.org/pub/projects/pgFoundry/dbsamples/usda/usda-r18-1.0/usda-r18-1.0.tar.gz
wget https://ftp.postgresql.org/pub/projects/pgFoundry/dbsamples/iso-3166/iso-3166-1.0/iso-3166-1.0.tar.gz

unzip pagila-0.10.1.zip
tar -xzvf iso-3166-1.0.tar.gz
tar -xzvf usda-r18-1.0.tar.gz

createdb pagila
createdb usda
createdb iso
createdb teste
createdb pgbench

psql -f iso-3166/iso-3166.sql iso
psql -f pagila-0.10.1/pagila-schema.sql pagila
psql -f pagila-0.10.1/pagila-data.sql pagila
psql -f usda-r18-1.0/usda.sql usda
pgbench -i -s 500 pgbench
```

pg_basebackup, pré requisitos

01 Aumentar o parâmetro max_wal_senders para comportar o backup (quando necessário

```
ALTER SYSTEM SET max_wal_senders TO 10;
```

02 Criar uma role de replicação

```
CREATE ROLE backup REPLICATION LOGIN;
```

03 Permitir conexão remota no pg_hba.conf para esta role a partir do servidor que vai realizar o backup

```
host replication   backup   pg01/32   scram-sha-256
host replication   backup   pg02/32   scram-sha-256
```

Backup c/ pg_basebackup (formato .tar.gz)

- Opções utilizadas:
  - -D destino do backup
  - -h Origem do backup
  - -U usuário
  - -p porta
  - -F Formato:
    - p = plano (padrão)
    - t = tar
  - -z comprimir no formato tar
  - -P Exibir informações do andamento do backup

Backup c/ pg_basebackup

- Exemplo com saída no formato .tar.gz

```
pg_basebackup -Ft -z -P -h pg01 -U backup -D /mnt/backup
```

- Exemplo com saída direta em $PGDATA:

```
pg_basebackup -Fp -P -h pg01 -U backup -D
/var/lib/postgresql/15/main
```

Restore no formato tar

- Criar um cluster (se não for utilizar o padrão)
```
pg_createcluster 15 main
```
- Baixar o serviço do PostgreSQL
```
systemctl stop postgresql
```
- Remover arquivos existentes do cluster
```
rm -Rf /var/lib/postgresql/15/main/*
```
- Descompactar o backup
```
cd /var/lib/postgresql/15/main
tar -xzvf /mnt/backup/base.tar.gz
cd /var/lib/postgresql/15/main/pg_wal
tar -xzvf /mnt/backup/pg_wal.tar.gz
```
- Subir o serviço
```
systemctl start postgresql
```

Restore no formato plain

- Criar um cluster (se não for utilizar o padrão)
  ```
  pg_createcluster 15 main
  ```
- Baixar o serviço do PostgreSQL
  ```
  systemctl stop postgresql
  ```
- Remover arquivos existentes do cluster
  ```
  rm -Rf /var/lib/postgresql/15/main/*
  ```
- Realizar o backup
  ```
  pg_basebackup -Fp -P -h pg01 -U backup -D
  /var/lib/postgresql/15/main
  ```
- Subir o serviço
  ```
  systemctl start postgresql
  ```

Point In Time Recovery

- Permite recuperar o backup num ponto no tempo específico
- Exige a criação de um arquivo `recovery.signal` para entrar no modo recovery antes de subir o serviço
- Exige que os **archives** estejam configurados e disponíveis

Archives

```
wal_level           = replica

archive_mode        = on

archive_command     = 'cp %p /mnt/archives/15/main/%f'

restore_command     = 'cp /mnt/archives/15/main/%f %p'
```

Archive para um servidor remoto:

```
archive_command       = 'scp %p pg02:/mnt/archives/15/main/%f'
```

Archives

- Forçando o arquivamento agora:

```
SELECT pg_switch_wal();
```

- Verificando o funcionamento:

```
postgres=# SELECT * FROM pg_stat_archiver ;
-[ RECORD 1 ]------+------------------------------
archived_count     | 0
last_archived_wal  |
last_archived_time |
failed_count       | 0
last_failed_wal    |
last_failed_time   |
stats_reset        | 2022-10-24 14:06:56.530887+00
```

Point In Time Recovery

```
recovery_target: 'immediate'

recovery_target_time: '2023-06-24 13:00:00-03'
```

Point In Time Recovery

Cenário 1: Restaurar até o último archive

- Rodar o pg_basebackup
- Realizar novas alterações na base após o último backup
- Criar o arquivo `$PGDATA/recovery.signal`
- Subir o serviço

Point In Time Recovery

Cenário 2: Restaurar até um ponto no tempo específico

- Rodar o pg_basebackup
- Realizar novas alterações na base após o último backup
- Anotar o horário atual
- Editar o parâmetro `recovery_target_time` para o horário anotado
- Criar o arquivo `$PGDATA/recovery.signal`
- Subir o serviço

Point In Time Recovery

Cenário 3: Restaurar até o primeiro ponto consistente

- Rodar o pg_basebackup
- Realizar novas alterações na base após o último backup
- Editar o parâmetro `recovery_target = 'immediate'`
- Criar o arquivo `$PGDATA/recovery.signal`
- Subir o serviço

Standby c/ pg_basebackup

01 Criar um slot de replicação (no master)

```
SELECT pg_create_physical_replication_slot ('standby_slot')
```

02 Rodar o pg_basebackup

```
pg_basebackup -P -R -h pg01 -U backup -S standby_slot -D
/var/lib/postgresql/15/main
```

03 Criar o arquivo standby.signal

```
touch $PGDATA/standby.signal
```

Promover o standby a master

- `pg_ctlculster 14 main` **`promote`** (Debian e derivados)
- `pg_ctl` **`promote`** (Demais situações)

Outros tópicos

- Estratégias de backup
- Backup lógico com pg_dump / pg_dumpall
- Outras ferramentas de backup físico
- Backup direto para block storage na nuvem

Obrigado!

fabio.telles@gmail.com

https://savepoint.blog.br

https://pgday.com.br

https://t.me/postgresqlbr